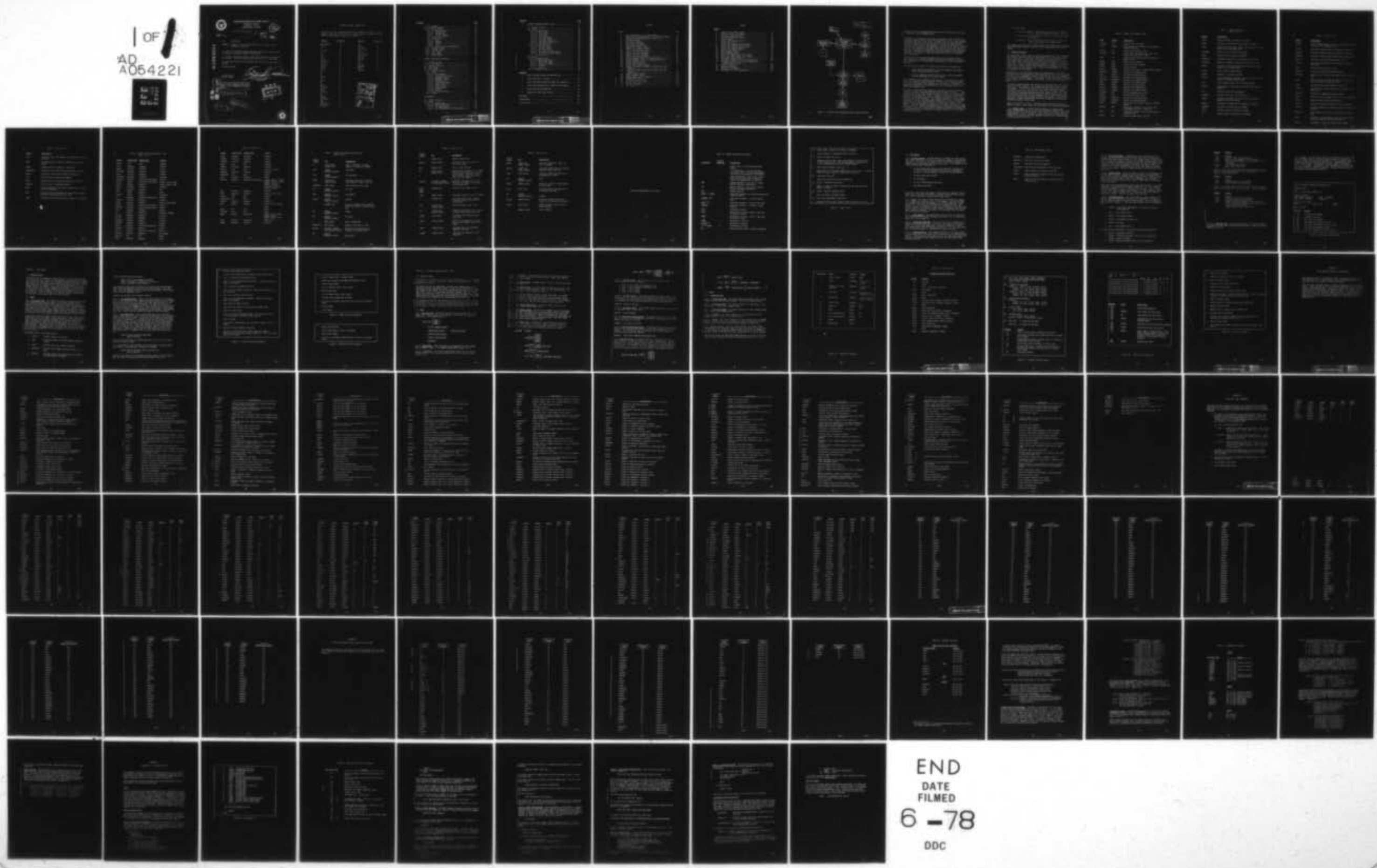
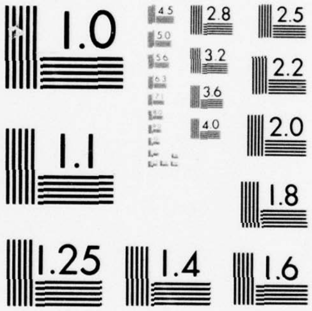
AD-A054 221 COMMAND AND CONTROL TECHNICAL CENTER WASHINGTON D C F/G 15/7
THE CCTC QUICK-REACTING GENERAL WAR GAMING SYSTEM. (QUICK). USE--ETC(U)
MAY 78

UNCLASSIFIED CCTC-CSM-UM-9-77-V1-CH-1 NL

1 OF 1
AD
A054221



END
DATE
FILMED
6-78
DDC


1.0
4.5
5.0
5.6
6.3
7.1
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

DEFENSE COMMUNICATIONS AGENCY

COMMAND AND CONTROL
TECHNICAL CENTER
WASHINGTON, D. C. 20301

(12)

IN REPLY
REFER TO: C314

V01 I A047256

NMCSSC

TO: RECIPIENTS

CCTC-

SUBJECT: Change 1 to Users Manual CSM UM 9-77, Volume I, Data Management Subsystem

1. Insert the enclosed change pages and destroy the replaced pages according to applicable security regulations.

2. A list of Effective Pages to verify the accuracy of this manual is enclosed. This list should be inserted before the title page.

3. When this change has been posted, make an entry in the Record of Changes.

FOR THE DIRECTOR

97 Enclosures
Change 1 pages

J. DOUGLAS POTTER
Assistant to the Director
for Administration

(6) The CCTC Quick-Reacting General War Gaming System. (Quick). Users Manual. Volume 1. Data Management Subsytem, Change 1.

(14) CCTC-CSM-UM-9-77-V1-CH-1

(12) 159 p.

409 658

EFFECTIVE PAGES - MARCH 1978

This list is used to verify the accuracy of CSM UM 9-77 Volume I after change 1 pages have been inserted. Original pages are indicated by the letter O, and change 1 pages by the numeral 1.

Page No.	Change No.
Title Page	0
ii-iii	0
iv	1
v	0
vi-viii	1
ix	0
1-2	0
3-4	1
5-22	0
23-26	1
26.1-26.2	1
27-29	1
29.1-29.2	1
30-40	0
41-43	1
44-48	0
49	1
50	0
51	1
52-58	0
59	1
60-82	0
83	1
84	0
85-86	1
86.1-86.2	1
87-92	0
93-94	1
94.1-94.2	1
95	1
96-97	0
98-106	1
106.1-106.4	1
107	0
108-116	1
116.1-116.2	1
117-121	0
122	1
123	0
124-126	1
126.1-126.2	1
127	0
128-130	1
130.1-130.2	1
131-139	0
140-141	1
142-143	0
144	1
145	0
146-152	1
153-154	0
155	1
156-158	0

Section Page

Section | Page

Appendix

FIGURES

TABLES

<u>Table</u>

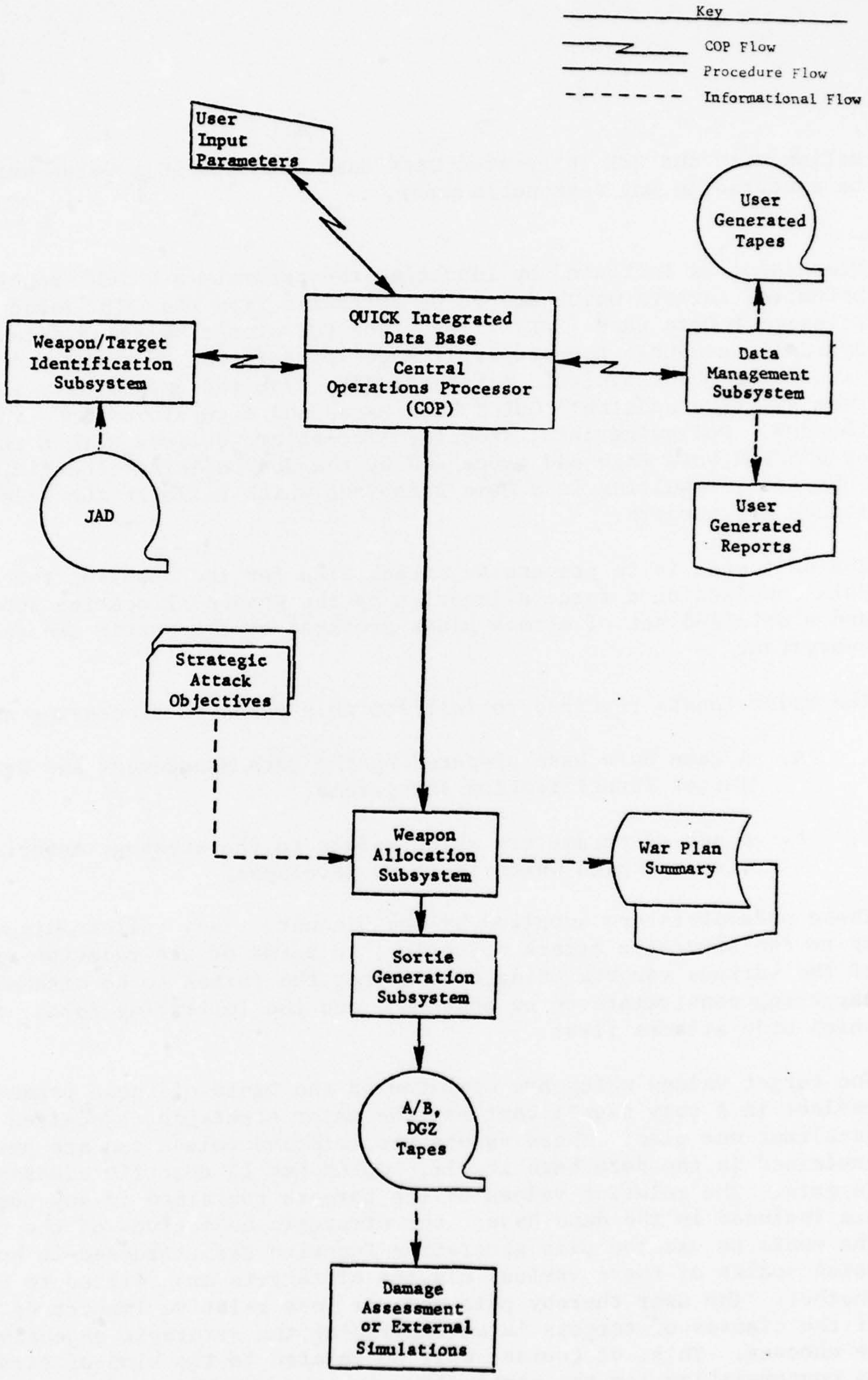

Key
COP Flow
Procedure Flow
Informational Flow
User Input Parameters
User Generated Tapes
QUICK Integrated Data Base
Central Operations Processor (COP)
Weapon/Target Identification Subsystem
Data Management Subsystem
JAD
User Generated Reports
Strategic Attack Objectives
Weapon Allocation Subsystem
War Plan Summary
Sortie Generation Subsystem
A/B, DGZ Tapes
Damage Assessment or External Simulations

Figure 2. Procedure and Information FLOW in QUICK/HIS 6000

action with the COP Integrated Data Base and related modules which may be executed in any reasonable order.

Processing is initiated by inputting the parameters which identify the potential targets which are to be extracted from the CCTC Joint Resource Assessment Data Base (JAD) files. The COP stores selected data and dynamically conducts the proper linkage for referral by other modules within the QUICK system. Alternatively, required target data is obtained from existing updatable QUICK Data Bases and also stored and linked by the COP. Following this, specified forces are defined within the developed QUICK Data Base and processed by the Weapon/Target Identification subsystem, resulting in a Game Data Base which reflects the selected forces and targets.

The next step is to prepare an attack plan for the opposing forces. This consists of a force allocation by the Weapon Allocation subsystem, and a detailed set of attack plans prepared by the Sortie Generation subsystem.

The major inputs required to initiate this phase of processing are:

a. A game data base prepared by the Data Management and Weapon/ Target Identification subsystems

b. A set of parameters which relate to the strategy associated with the plan which is to be developed.

These parameters are supplied by the planner. They reflect his views as to the strategic attack objective, in terms of the relative values of the various targets being considered, the forces to be withheld, the targeting constraints to be observed, and the initiating force, i.e., which side attacks first.

The target values which are computed on the basis of these parameters reflect in a very significant way the major strategic objectives of the resultant war plan. These values are relative values and are partially contained in the data base itself. QUICK has 15 specific classes of targets. The relative values of the targets contained in any one class are included in the data base: the strategic objectives of the planner who wants to use the plan generation function are expressed in how the value scales of these various classes of targets are related to one another. The user thereby puts more or less relative importance on each of the classes of targets in accomplishing the strategic objectives that he chooses. This, of course, will be related to the kind of strategy he is contemplating for the particular war game, whether a first or second strike, and so forth.

b. Phrase Types

- Relational Phrases - Sublevels are used to define comparative or logical meaning according to operators
- Restricted Relational Phrases - Used to set attributes and/or variables
- Elements - Used to set single elements such as value elements, special words or display names.

The clause types define the overall syntax of the clause, and the phrase types define which phrases fall into or are connected by the overall syntax.

3.3 QUICKs Dictionary

All permissible words within the developed language that have meaning to the COP are contained within the 'Dictionary' (a list of tables stored in the data base). These words contain attributes (as employed for target and weapon definition), plus other words necessary for the syntax. Words within the dictionary are grouped as outlined in previous subsections. Based on these groupings, tables 6 through 11 present the entire list of words defined within QUICKs command language plus comments on how each word may be used. The list of attributes are defined in appendix A of this manual.

Occasionally a user will wish to enter an alphanumeric string which ordinarily will be recognized by the dictionary but which the user wishes to treat as an alphanumeric constant. The user may do so by enclosing the string in quotation marks. For example, if the user wishes the string AS to be treated as alphanumeric rather than a null, he should enter it thus - "AS".

A sentence command written with entries not contained within the cited tables are words foreign to the language and may cause computer execution aborts. In addition, only certain combinations of words from the "dictionary" (such as verbs, adverbs, etc.) have structural meaning. The entire QUICK system generates its final output through a series of selections of individual program modules as defined by the verb. The selected module then can act (or interpret) only on those sentence patterns that request outputs produced within the program. This is also analogous to formal English where individual thoughts are expressed in separate paragraph constructions.

Many of the Adverbs in table 7 introduce clauses which are used by a number of verbs. These clauses are described in the following paragraphs.

3.3.1 DEFINE Clause. A DEFINE clause consists of a single equals phrase in which the subject is used as an alphabetic variable name. (The subject should not be an attribute.) The object of the phrase should be a mathematical formula combining attributes and numeric values plus any alphabetic string which the user intends to employ as the variable name of this or any other DEFINE.

Table 6. QUICK's Text English Verbs

VERB	MODULE	DESCRIPTION
ALLOCATE	ALOC	Executes the ALOC module
ALTER	REPORT	Makes updates and changes to a previously designed display
ASSIGN	JLM	Builds the Assignment Table
ASTERISK	JLM	Makes Damage Assessment Tape from a JAD format tape and the Data Base
BUILD	EIM	Instructs the External Interface Module to build a file
CHANGE	DATA	Updates data element(s)
CREATE	DATA	Creates new data elements
DELETE	DATA	Deletes records
DESIGN	REPORT	Constructs and saves a REPORT module display
DGZSELECT	ALOCOUT	Executes the ALOCOUT module
EDIT	EDITDB	Executes the EDITDB module
EVALUATE	EVALALOC	Executes the EVALALOC module
FOOTPRNT	FOOTPRNT	Executes the FOOTPRNT module
INDEX	INDEXER	Executes the INDEXER module
MODIFY	DBMOD	Executes the BDMOD module
PLANOUT	PLANOUT	Executes the PLANOUT module
PLANSET	PLANSET	Executes the PLANSET module
PLOTDATA	EIM	Creates geography plot tapes
PLOTIT	EIM	Creates sortie plot tapes
POSTALOC	POSTALOC	Executes the POSTALOC module
PREPARE	PREPALOC	Axecute the PREPALOC module
PRINT	REPORT	Prints some user defined (built by DESIGN) display
RESTORE	SRM	Brings the contents of an IDS data base from magnetic tape to a disk file
SAVE	SRM	Stores the contents of an IDS data base on a magnetic tape
SELECT	JLM	Selects records from a JAD file

Table 7. QUICKs Adverb List
(Part 1 of 3)

ADVERB	DESCRIPTION
ABTAPE	Requests PLANOUT to produce an AB tape
ACARD	Describes a non-MIRV missile sortie to PLANOUT
ALPHAS	Defines the CLASS, TYPE, DESIG, and category relationships for the Assignment Table
ATTACKERS	Defines the attacking weapon systems
CCARD	Describes sortie change to PLANOUT
COUNTRIES	Introduces countries to be processed by EVALALOC
DEFENDERS	Defines the exemplar targets and associated values
DEFINE	Describes a user defined variable for use in either printed or tape/file output
DISPLAY	Names a REPORT module display
EQUATE	Describes a footprint equation
FIELDS	Lists attributes in the data base to be tested to see if they are within defined ranges
FILE	Used to define type of file EIM should build
FINDMIN	Gives number of iterations for FINDMIN subroutine of ALOCOUT
FIX	Defines weapon fixed assignments
FLAGREST	Describes allocator flag restrictions
FORMAT	Describes the format to be used in the creation of printed or tape/file output
FUNCOM	Introduces the function and commands for PLANOUT
GAMETIME	Introduces the game time for PLANOUT
ICARD	Describes insert in sortie to PLANOUT
IF	Defines limits for equations in FOOTPRNT

Table 7. (Part 2 of 3)

ADVERB	DESCRIPTION
KEEPING	Lists the DESIGs which are not to be deleted from the data base after ASTERISK is run
LOCREST	Describes allocator country location restrictions
MINRANGE	Describes modifications to the RNGMIN attributes
MIRVREST	Describes allocator MIRV weapon system restrictions
MISTME	Describes time coordinated missile attacks
MODRANGE	Describes modifications to the RANGE and RANGEREF attributes
MSLCOR	Describes missile launch timing
ONPRINTS	Defines user controlled printed output
OMITTING	Keeps duplicate targets from being added to the data base
ORDER	Allows the user to input the order that the classes will be entered into the data base
PLANE	Introduces plane-type indexes for PLANOUT
PLAYERS	Describes valid country codes and the regions that they are in
PRIORITY	Introduces a complexing priority scheme
PUNCH	Produces cards containing final allocator multipliers
READMUL	Reads in pre-set allocator multipliers
RECALC	Indicates recalculation mode for various modules
REEQUATE	Describes changes to a previously entered footprint equation
REPLACING	Causes duplicate targets to be replaced in the data base
SAME	Describes a record similar to the one being created which is to be used for default values.
SETTING	Introduces a clause that stores data elements

Table 7. (Part 3 of 3)

ADVERB	DESCRIPTION
SMAT	Describes values and updates for the SMAT array of the allocator.
SORT	Describes the sort order for printed or tape/file output
STRIKE	Request PLANOUT to produce a Strike tape
SUPRESSING	Suppresses data value editing during data creation
TGTMOD	Introduces target value changes for EVALALOC
UNIT	Gives tape/file logical unit number
USING	Requests data value editing during data creation
VNOPTION	Used to select a complexing option
WHERE	Describes subset of data base on which verb's action is to be performed
WITH	Describes the relationships that must be met between selected attributes in the data base
VPMOD	Introduces weapon characteristic changes for EVALALOC

Table 8. QUICKs Text English Adverbs Usage
(Part 1 of 2)

ADVERB	CLAUSE TYPE	PHRASE TYPE	VERB(S)
ABTAPE	Null	Null	PLANOUT
ACARD	Sequence	Elements	PLANOUT
ALPHAS	Sequence	Elements	ASSIGN
ATTACKERS	Sequence	Elements	PLANSET
CCARD	Sequence	Elements	PLANOUT
COUNTRIES	Sequence	Elements	EVALUATE
DEFENDERS	Sequence	Restricted Relational	PLANSET
DEFINE	Single	Restricted Relational	DESIGN, ALTER, BUILD
DISPLAY	Sequence	Elements	DESIGN, ALTER, PRINT
EQUATE	Sequence	Elements	FOOTPRNT
FIELDS	Sequence	Elements	EDIT
FILE	Single	Element	BUILD
FINDMIN	Single	Element	DGZSELECT
FIX	Sequence	Restricted Relational	PREPARE
FLAGREST	Sequence	Elements	ALLOCATE
FORMAT	Sequence	Elements	DESIGN, ALTER, BUILD
FUNCOM	Sequence	Restricted Relational	PLANOUT
GAMETIME	Sequence	Restricted Relational	PLANOUT
ICARD	Sequence	Elements	PLANOUT
IF	Boolean	Relational	FOOTPRNT, PLANOUT
KEEPING	Sequence	Elements	ASTERISK
LOCREST	Sequence	Elements	ALLOCATE
MINRANGE	Sequence	Elements	ALLOCATE
MIRVREST	Sequence	Elements	ALLOCATE
MISTME	Sequence	Restricted Relational	PLANOUT
MODRANGE	Sequence	Elements	ALLOCATE
MSLCOR	Sequence	Restricted Relational	PLANOUT
ONPRINTS	Sequence	Elements	(ALL VERBS)
OMITTING	Null	Null	SELECT
ORDER	Sequence	Elements	SELECT

Table 8. (Part 2 of 2)

ADVERB	CLAUSE TYPE	PHRASE TYPE	VERB(S)
PLANE	Sequence	Elements	PLANOUT
PLAYERS	Sequence	Elements	ASSIGN
PRIORITY	Sequence	Elements	PLANSET
PUNCH	Null	Null	ALLOCATE
READMUL	Sequence	Elements	ALLOCATE
RECALC	Null	Null	ALLOCATE, PLANOUT
REEQUATE	Sequence	Elements	FOOTPRNT
REPLACING	Null	Null	SELECT
SAME	Sequence	Elements	CREATE
SETTING	Sequence	Restricted Relational	PLANET, MODIFY, DESIGN, PREPARE, CREATE, ALTER, CHANGE, PLOTDATA, SELECT, ALLOCATE, EVALUATE, POSTALOC, PLOTIT, PLANOUT
SMAT	Sequence	Elements	ALLOCATE
SORT	Sequence	Elements	DESIGN, ALTER, BUILD
STRIKE	Null	Null	PLANOUT
SUPPRESSING	Null	Null	CREATE
TGTMOD	Sequence	Elements	EVALUATE
UNIT	Single	Element	SELECT, SAVE, RESTORE, BUILD
USING	Null	Null	CREATE
VNOPTION	Null	Null	INDEX
WHERE	Boolean	Relational	DESIGN, PRINT, ALTER, CHANGE, DELETE, SELECT, BUILD, EDIT
WITH	Sequence	Relational	INDEX, EDIT
WPNMOD	Sequence	Elements	EVALUATE

Table 9. QUICKs Text English Special Words (Part 1 of 3)

SPECIAL WORD	USE	DESCRIPTION
A	SORT adverb PLANOUT verb	Same as ASCENDING for SORT, Implies air burst for PLANOUT
ACOS	EQUATE, REEQUATE adverbs	Arc-cosine
ACOT	EQUATE, REEQUATE adverbs	Arc-cotangent
AFTER	ALTER verb* FORMAT adverb	Introduces additions to format after indicated PAGE, LINE, etc.
ASCENDING	SORT adverb	Lowest values will be first
ASIN	EQUATE, REEQUATE adverbs	Arc-sine
ATAN	EQUATE, REEQUATE adverbs	Arc-tangent
AZIMUTH	EQUATE, REEQUATE adverbs	Azimuth
C	PLANOUT verb	Value for "CALOFF" field, implies DLATOF and DLONGOF represent actual DGZ
COS	EQUATE, REEQUATE adverbs	Cosine
COT	EQUATE, REEQUATE adverbs	Cotangent
D	SORT adverb	Same as DESCENDING
DESCENDING	SORT adverb	Highest values will be first
EXCLUDE	FLAGREST, LOCREST, and COUNTRIES adverb	Indicates following items are excluded from consideration
EXP	EQUATE, REEQUATE adverbs	Exponential

Table 9. (Part 2 of 3)

SPECIAL WORD	USE	DESCRIPTION
G	PLANOUT verb	Implies ground burst
HEADER	FORMAT adverb	Introduces title that appears at top of each page
HHR	PLANOUT verb	Start of game (H-Hour)
IN	SELECT verb; FORMAT adverb	Restricts the assignment of a target based on the country it is located in for SELECT. Introduces user specified data item output format (e.g., I4) for FORMAT
INCLUDE	FLAGREST, LOCREST, and COUNTRIES adverbs	Indicates following items to be included, any others are excluded from consideration
KDAY KMON KYEAR	PLANOUT verb	Values for day, month and year
LINE	FORMAT adverb	Introduces format for body of report
N	PLANOUT verb	Value for "RAC" field. Implies attrition factors should not be recalculated
NEW	DESIGN verb, DISPLAY adverb	Indicates display name is new
OLD	DESIGN verb, DISPLAY adverb	Indicates new display is to replace previous display with same name
OTHER	FILE adverb	Introduces user defined output tape file
OWNED	ALPHAS adverb	Restricts assignment of a target bases on the country code of its owner
PAGE	FORMAT adverb	Introduces new set of HEADERs, TRAILERs, and LINEs
PAGENO	FORMAT adverb	Indicates page number is to be displayed

Table 9. (Part 3 of 3)

SPECIAL WORD	USE	DESCRIPTION
REMOVE	ALTER verb, FORMAT adverb	Indicates named PAGE, LINE, etc. is to be removed
REPLACE	ALTER verb, FORMAT adverb	Introduces format items to replace indicated PAGE, LINE, etc.
SIDAC	FILE adverb	Indicates output tape/file is to be produced for SIDAC
SIN	EQUATE, REEQUATE adverb	Sine
SPACES	FORMAT adverb	Indicates a series of blank spaces in the format
TABLE	FILE adverb	Indicates output tape/file is to be produced for TABLE
TAN	EQUATE, REEQUATE adverb	Tangent
TRAILER	FORMAT adverb	Introduces format which is to appear at the bottom of each page
UPDATE	SMAT adverb	Indicates input values are permanent updates
X	FORMAT adverb	Same as SPACES

THIS PAGE INTENTIONALLY LEFT BLANK

Table 10. QUICKs Text English Operators

ALPHABETIC	SPECIAL CHARACTER	DESCRIPTION
	,	Comma - used in collections and other formats
	(	Left Parenthesis - used to introduce collections and for associative purposes in boolean and mathematical expressions
	)	Right Parenthesis - used to end collections and for associative purposes in boolean and mathematical expressions
AND	&	Boolean operator - also used to extend EQUALS relational expressions
OR		Boolean operator - also used to extend EQUALS relational expressions
NOT		Boolean operator
EQUAL or EQUALS	=	Relational operator (two spellings permitted: EQUAL or EQUALS)
GREATER THAN	>	Relational operator - strictly greater than
LESS THAN	<	Relational operator - strictly less than
BETWEEN		Relational operator - defines a closed interval
LIKE		Relational operator
PLUS	+	Mathematical operator (symbol + may also be used as a sign)
MINUS	-	Mathematical operator (symbol - may also be used as a sign)
TIMES	*	Mathematical operator
DIVIDED BY	/	Mathematical operator
TO THE POWER	**	Mathematical operator
OF		Special operator used to modify attributes

21 SYNTAX ERROR - RELATION NOT RIGHT IN SEQUENCE

Normal sequence of relational phrase violated.

22 TYPE (I5) VALUE (I5) (A12)

Arguments are the type, value and alphabetic representation of the offending item. This message appears following most of the messages that begin with "SYNTAX ERROR"

23 INPUT TABLES EXCEEDED, TYPE (I2)

Tables built by subroutine ERRFND have been exceeded. Argument indicates type of table in which error occurred.

24 LONG STRING TOO LONG

Input long string exceeds 120 characters.

25 INPTRN - UNBALANCED COLLECTION

Number of items in object collection(s) does not agree with number in subject.

26 INPTRN - BOOLEAN STATEMENT WRONG

Error in boolean statement logic.

27 $*$*$ (A6) ENCOUNTERED ERROR $*$*$

Agreement is link name of module which encountered an error.

Figure 5. (Part 3 of 3)

4.4 BOOT Module

4.4.1 General Purpose. The BOOT Module is designed to create and update those portions of the data base which are essential to normal COP operation. As a result, the input to BOOT is on fixed formatted card images rather than a free form text English input. The portions of the data base which BOOT addresses are:

- o The data organization index, which contains a functional description of the IDS data base strucutre, the information required to retrieve headers and the data editing directory
- o The data entry point headers
- o The dictionary
- o The text English syntax directory
- o The module link table

In general, each input card image to BOOT creates or updates a record within one of the above structures. Normally, BOOT will be run only when a QUICK data base is being created from scratch.

4.4.2 Input. The input to BOOT consists of an introductory command verb followed by a series of sets of card images. Each set is begun by a command adverb and terminated by a card containing END in card columns 1-3. The last set is followed by a second END card. The order of the sets is important if the user is building a data base from scratch because the creation of some records is dependent upon the previous creation of others. The order to be followed is that in which the sets will be presented.

4.4.2.1 Verb Command. The introductory verb for BOOT is INITIALIZE. This command must be defined starting with column 1 on the first card image that the COP reads in.

4.4.2.2 Introductory Adverbs. As cited, each set is introduced by a command adverb and ended with an END card. The command adverbs must appear on separate card images with the adverb starting in column 1. The command adverbs and the general description of the sets they introduce are shown in table 12. Discussions of each set follow.

4.4.2.3 NEWINDEX Adverb. This command adverb has no following cards. It must appear only in the case of a construction of the data base from scratch. In this case it must appear first. It causes the creation of the data organization index header and a utility table header used internally by COP.

Table 12. BOOT Command Adverbs

NEWINDEX	- Creates new index header
RECORDTYP	- Adds new record type records
HEADER	- Adds data entry point headers
DICTIONARY	- Makes entries and updates to the dictionary
MODULE	- Makes changes to the module link table
SYNTAX	- Makes entries and changes to the text English syntax directory
INDEX	- Makes entries and changes to the data organizational index

4.4.2.4 RECORDTYP Adverb. A complete and total definition of each record type contained within the integrated data base is presented in Program Maintenance Manual Volume I. This command adverb introduces the list of those record types to be added to the data organizational index. Each card image following the adverb contains the name and number of the record type. For each card image, the record type name is placed in columns 1 to 8 (left justified) and the record type number in columns 9 to 16 (free field). The set is ended normally with the insert of the END card.

4.4.2.5 HEADER Adverb. In the IDS sense, a header is the highest level of entry into a series of master records and their associated chains. In the QUICK data base, headers define entries to data elements through the use of attribute CLASS. This set of commands, then, specifies header name (or class entries) and, further, links them to record type as outlined above.

Each card in the set defines the record type name (columns 1 to 8, left justified), the attribute CLASS name (columns 9 to 16), the attribute SIDE name (columns 17 to 24). SIDE may be left blank. Also, if the header record type is TGTHD (target header), the card image may contain an entry for ICLASS (columns 25 to 32) which is used as an internal sequential counter. This set is ended by an END card.

4.4.2.6 DICTIONARY Adverb. All of the words that have meaning to the COP are contained within the dictionary (see appendixes A, B, and C) and are added or altered with this command adverb. The card image format is shown in figure 6. The word types defined with the dictionary are:

o OPERAT - text English mathematical, relational, boolean or syntax operator

o VERB - text English verb

o ADVERB - text English adverb

o SPCIAL - text English special word

o ATTRIB - attribute

o NULL - text English null

If the word is an attribute, the various type identifications are:

o NMALPH - alphabetic attribute

o IDALPH - alphabetic attribute used as an IDS identifier

o NMNUMR - numeric attribute

o IDNUMR - numeric attribute used as an IDS identifier

Columns	Meaning
1-6	=INDATR
9-24	Attribute Name (left justified)
25-32	Attribute type mnemonic (left justified)
33-40	Default value (alphabetic values should be left justified
41-48	Lower limit (free field)
49-56	Upper limit (free field)

d. ALPHVL -- This card image creates the list of legal values for attributes whose type mnemonic is some combination of 'LIST' values. Each card image is:

Columns	Meaning
1-6	=ALPHVL
9-24	Attribute name (left justified)
25-32	'LIST' values (left justified)

e. LINKER -- This card image is used to inform the data organization as to the make up of each record type. The card image is:

Columns	Meaning
1-6	=LINKER
9-16	Record type name (left justified)
17-32	Attribute name (left justified)
33-40	Control identifier mnemonic (left justified) =CNTROL, attribute used as a match key for internal structure definition =NORM, attribute not a control one

4.4.2.11 Final BOOT Card. After the last input set to BOOT has been terminated, the user must include another END card. Normal text English commands may follow.

4.4.3 Output. The BOOT module produces an annotated list of its input on print report code 13. Each card image as input is displayed as input. (Columns 73-80 are also printed so as to provide any sequencing information the user desires to place there.) Command adverbs are given a special flag. In addition, each card image is preceded by a +, @ , or $ symbol. Respectively, these symbols imply a record type was added, altered or an error occurred. A sample of portion of the BOOT module output appears in figure 7.

```
①***** CENTRAL OPERATIONS PROCESSOR *****

②INITIALIZE

③***** COLD BOOT PROCESS *****

④***** SYNTAX
⑤@ PRMADV  OMITTING         BOOL    ELEMENT
⑥$ SYNCLZ  DESIGN           ONPRINTS

④***** DICTIONARY
⑦+ SIZE             ATTRIB  338      0        NMNUMB

④***** END
```

HEADING	MEANING
①	COP input print header
②	Command to run BOOT
③	BOOT Input Print Header
④	BOOT Input set introductory card
⑤	This card caused BOOT to alter a record
⑥	This card was rejected by BOOT
⑦	This card caused BOOT to add a record

Figure 7. BOOT Module Output Print

SECTION 5. DATA MODULE

5.1 General Purpose

The DATA module is designed to allow the user to create those portions of the data base not created by other modules, add information to existing record types which were only partially initialized by other modules, make corrections and updates to existing record types and delete record types which are no longer desired. The DATA module has three verbs: CREATE, CHANGE, and DELETE. The general philosophy of the DATA module is to allow the user to make commands solely in terms of attributes, relieving him of the necessity of knowing which record type or types are affected. Also, the execution of the DATA module is permissible at any stage of processing within the QUICK system.

5.2 Input

5.2.1 The CREATE Verb. The CREATE verb defines new data record types. The DATA module builds record types to be created filling in any unsupplied attribute values either from QUICKs directory defaults or from a user specified source. In general, the use of the CREATE verb will cause one or more sets of record types to be created. DATA also will check all input values against QUICKs directory for validity. Finally, the DATA module checks to make sure that no duplicate records are created such as two separate target records having the same DESIG.

If the user desires to create targets via DATA he does not necessarily have to specify values for attributes TYPE, DESIG, TASK, or any other attribute value normally obtained if the JLM has successfully built an Assignment table (see Users Manual Volume II, section 2). These values will be supplied from the Assignment table where not user specified. (Note that under certain options JLM can run successfully without an Assignment table. In these cases the user must supply all data and is responsible for the uniqueness of the value for the DESIG attribute.)

The CREATE verb has five legal adverbs:

- o SETTING - defines values to be stored
- o SAME - alternate method of storing default attribute values
- o USING - specifies the use of QUICKs directory
- o SUPRESSING - specifies that QUICKs directory is not to be used
- o ONPRINTS - optional request for printing data of interest to a maintenance programmer

A sample CREATE verb input could be:

```
CREATE SETTING CLASS=BOMBER SIDE=BLUE
(IREG, LAT, LONG) = (1,65,150) & (2,63,51) &
(3,63,3) ONPRINTS USING DIRECTORY
```

This command creates three records for the given CLASS and SIDE where each record has the input values for IREG, LAT and LONG. Note that TYPE, TASK, and CNTRYLOC are not given; they are assumed to be obtained from a created Assignment table.

Comments on the major use of adverbs follows.

5.2.1.1 The SETTING Adverb. This is the chief clause for the CREATE verb. It is that part of the command structure whereby the user informs the COP what values are to be stored and under what record type. Any attribute may be set through this adverb. One setting clause will create one record type except if the clause contains one or more extended equals phrases (i.e., LAT=40 & 50 & 60) in which case a record type set will be created for every combination of the values in such phrases. Note that a create verb may contain any number of SETTING clauses.

5.2.1.2 The SAME Adverb. This clause allows the user to specify a logical record which is used to define the default values for attributes which the user has not input. Its purpose is to reduce the inputting of redundant information. For instance, say the user desires to create two weapon systems, one for a B52 and another for a B52A and both systems have identical attribute values except that the attribute RANGE differs between the two. Under this scenerio, the user would have to input two complete sets of identical attribute values if not for the SAME clause. Assume the B52 record has been created and all of its attributes stored. A command, then, to create the B52A record could be:

```
CREATE SETTING TYPE=B52A RANGE=9000
       SAME AS TYPE B52
```

Now the B52A record type is created identically to the B52 but with a different value for RANGE.

It is appropriate to show examples of the LIKE and OF operation which are helpful in creating data. Consider the command:

```
CREATE SETTING TYPE=B52A RANGE LIKE TYPE F111
       SAME AS TYPE B52
```

Type B52A will have all of the same attribute values as the B52 except that the RANGE for the B52A will equal the range of the F111.

1 SETTING CLAUSE WRONG FOR CREATE

An error has occurred in the setting clause, check inputs.

2 (F15.4) VIOLATES EDIT RANGE FOR (A12)

Edit error for floating point attribute. Value and attribute name are displayed.

3 (I10) VIOLATES EDIT RANGE FOR (A12)

Edit error for integer attribute: value and attribute name are displayed.

4 (A6) IS NOT IN EDIT LIST FOR (A12)

Edit error for alphabetic attribute: value and attribute name are displayed.

5 DATA//ERROR IN SAME AS CLAUSE

DATA could not find record identified in SAME clause.

6 ERROR IN TARGET INPUT

Error detected by Assignment table. Not sufficient amount of data supplied or the wrong data supplied.

7 LINKUP FAILURE

User supplied insufficient data to build a viable retrieval scheme.

8 DUPLICATE STORE ATTEMPTED (2A7,2I10)

Fields are record type name, DESIG, IREG and ICOMPL. Message occurs when store attempt results in a D01 error--this error is not fatal.

Figure 11. CREATE Verb Error Messages

1 ERROR IN CHANGE VERB - MISSING CLAUSE

CHANGE verb requires both WHERE and SETTING clauses.

2 SETTING CLAUSE WRONG

Error in SETTING clause, check inputs.

3 WHERE CLAUSE WRONG

Error in WHERE clause, check inputs

4 NO RECORD TYPES DETERMINED IN CHANGE

User has not specified sufficient data to build a retrieval scheme.

5 LINKUP FAILURE

Same as above.

Figure 12. CHANGE Verb Error Messages

1 ERROR IN DELETE VERB

WHERE clause either in error or missing.

2 CANNOT BUILD SCHEME

User has not specified sufficient data to build a retrieval scheme.

Figure 13. DELETE Verb Error Messages

SECTION 9. EXTERNAL INTERFACE MODULE (EIM)

9.1 General Purpose

The purpose of the EIM is to crease output tapes/files which are designed to be input to external processors. EIM has three command verbs: BUILD, PLOTDATA and PLOTIT.

The BUILD verb has the capability to create two standard tapes plus a generalized output capability similar to that of the REPORT module. The user specifies the output file to be developed by inserting the special word SIDAC, TABLE, or OTHER within the BUILD command. The SIDAC word causes the production of two JAD format data base assessment tapes. The TABLE word causes the production of a standard set of tables. The OTHER word provides the capability to produce generalized data files.

The PLOTDATA and PLOTIT verbs produce an output tape suitable for use on the CALCOMP plotter. The plotable information includes penetration and depenetration corridors, refuel points and recovery bases where PLOTDATA is concerned, and bomber and tanker sorties where PLOTIT is used.

9.2 Input

9.2.1 The BUILD Verb. The BUILD verb has six optional adverbs: UNIT, WHERE, SORT, DEFINE, FORMAT, and ONPRINTS. It also has one required adverb FILE. The general form is:

```
                  ⎧ TABLE ⎫
BUILD FILE        ⎨ SIDAC ⎬
                  ⎩ OTHER ⎭

. . . [ON UNIT numeric-value]

. . . [WHERE where-clause] . . . [SORT sort-clause]

. . . [DEFINE define-clause]

. . . [FORMAT format-clause]

. . . [ONPRINTS]
```

9.2.1.1 FILE TABLE. This clause may be accompanied by a UNIT, WHERE, and/or ONPRINTS clause. The standard output unit defaults to 35.

9.2.1.2 FILE SIDAC. This clause stands alone except for the optional ONPRINTS clause. BLUE targets are defined on unit 35 and RED targets on unit 36.

9.2.1.3 FILE OTHER. Generalized data files are produced through user input.. A FORMAT clause must be used and UNIT, WHERE, SORT, DEFINE, ONPRINTS may be included.

9.2.1.4 The UNIT Adverb. The UNIT clause is used to specify an output unit other than 35.

9.2.1.5 The WHERE Adverb. For FILE TABLE or SIDAC instructions, this clause may only be used to specify a value for the SIDE attribute. If omitted the value of SIDE defaults to BLUE.

For FILE OTHER this clause is used to specify the subset of the data base to be accessed to build the output file. This clause is fully detailed in section 3 and examples of usage within the REPORT module, which is similar to the EIM, are presented in section 8. Normal DEFINE variables may be used. However, the OF phrase is not permitted.

9.2.1.6 SORT and DEFINE Adverbs. The use of these clauses are as given for the REPORT modules (section 8).

9.2.1.7 The FORMAT Adverb. The use of the FORMAT clause is different in the EIM from REPORT (section 8) in that headers, trailers and PAGNO must not be used. Furthermore the PAGE special word has a different meaning than explained elsewhere. When it encounters the word PAGE, the EIM writes an end-of-file mark on the output unit. (The user should declare the output tape as a multi-file.)

9.2.2 The PLOTDATA Verb. The PLOTDATA verb has one adverb SETTING. This adverb is followed by a series of equal relational phrases. Each phrase sets the values for a particular portion of the plot information. The general form of the PLOTDATA verb is as follows:

$$\underline{\text{PLOTDATA}} \quad \underline{\text{SETTING}}$$

$$\ldots \quad \underline{\text{MAP}} \left\{ \begin{array}{c} \underline{\text{EQUAL}} \\ \underline{=} \end{array} \right\} \left\{ \begin{array}{c} \underline{\text{PIC1}} \\ \underline{\text{PIC2}} \\ \underline{\text{PIC3}} \\ \underline{\text{PIC4}} \\ \underline{\text{PIC5}} \end{array} \right\}$$

$$\ldots \quad \underline{\text{SIDE}} \left\{ \begin{array}{c} \underline{\text{EQUAL}} \\ \underline{=} \end{array} \right\} \left\{ \begin{array}{c} \text{side} \\ \underline{\text{BOTH}} \end{array} \right\} [\underline{\text{AND}}\ \underline{\text{side}}]$$

$$\ldots \quad \underline{\text{SCALE}} \left\{ \begin{array}{c} \underline{\text{EQUAL}} \\ \underline{=} \end{array} \right\} \underline{\text{numeric-value}}$$

$$\ldots \quad \underline{(}\underline{\text{LAT}}\underline{,}\ \underline{\text{LONG}}\underline{)} \left\{ \begin{array}{c} \underline{\text{EQUAL}} \\ \underline{=} \end{array} \right\} \underline{(}\underline{\text{latitude}}\underline{,}\ \underline{\text{longitude}}\underline{)}$$

$$\ldots\quad \underline{(\text{PLOT},}\ \underline{\text{LABEL})} \left\{\begin{matrix}\underline{\text{EQUAL}}\\ \underline{=}\end{matrix}\right\} \left(\left\{\begin{matrix}\underline{\text{PENCOR}}\\ \underline{\text{DEPCOR}}\\ \underline{\text{RECOV}}\\ \underline{\text{REFUEL}}\end{matrix}\right\} \underline{,} \begin{matrix}\underline{\text{YES}}\\ \underline{\text{NO}}\end{matrix}\right) [\underline{\text{AND}}\ \ldots]$$

9.2.2.1 The MAP Phrase. This is used to set the type of plot. The user selects one of the following:

- o PIC1 - a PIC-1 (western hemisphere) plot
- o PIC2 - a PIC-2 (eastern hemisphere) plot
- o PIC3 - a 50 x 40 plot
- o PIC4 - a 20 x 40 plot
- o PIC5 - a 10 x 10 plot

9.2.2.2 The SIDE Phrase. This phrase selects the side or sides to be plotted. If the user wishes only one side plotted he states the side. If he wishes both sides plotted on the same plot he uses the value BOTH. If he wishes the sides plotted separately he uses an AND connector.

EXAMPLE: SIDE=BLUE AND RED

9.2.2.3 The SCALE Phrase. This phrase allows the user to set the ratio of real world units to map units.

EXAMPLE: SCALE=19010000

9.2.2.4 The Origin Setting Phrase. This phrase consists of the assignment of values to the attributes LAT and LONG. These values are to be used as the origin of the plot.

EXAMPLE: (LAT,LONG)=15.0,10.0)

9.2.2.5 The Plot Selection Phrase. This phrase details which data elements are to be plotted plus whether or not those elements are to be labeled. The equal phrase may be continued via the AND connector if multiple element types are desired.

EXAMPLE: (PLOT,LABEL)=(PENCOR,YES)AND(RECOV,NO)

9.2.3 The PLOTIT Verb. The PLOTIT verb has a similar form to the PLOTDATA verb. The SETTING clause will contain a MAP phrase, a SCALE phrase, and an "origin setting" phrase, each of which has the same application as in the PLOTDATA verb (the SIDE and plot selection phrases are not used). In addition, there is a SORTNO phrase which specifies tje sortie sequence numbers of desired sorties. The general form is:

$$\underline{\text{PLOTIT}}\ \underline{\text{SETTING}}\ \underline{\text{MAP}} \left\{\begin{matrix}\underline{\text{EQUAL}}\\ \underline{=}\end{matrix}\right\} \left\{\begin{matrix}\underline{\text{PIC1}}\\ \underline{\text{PIC2}}\\ \underline{\text{PIC3}}\\ \underline{\text{PIC4}}\\ \underline{\text{PIC5}}\end{matrix}\right\}$$

$. . . \ \underline{\text{SCALE}} \ \left\{\begin{matrix}\underline{\text{EQUAL}}\\ \underline{=}\end{matrix}\right\} \ \underline{\text{numeric value}}$

$. . . \ \underline{(}\ \underline{\text{LAT}}\ \underline{,}\ \underline{\text{LONG}}\ \underline{)} \ \left\{\begin{matrix}\underline{\text{EQUAL}}\\ \underline{=}\end{matrix}\right\} \ \underline{(}\ \underline{\text{latitude}}\ \underline{,}\ \underline{\text{longitude}}\ \underline{)}$

$. . . \ \underline{\text{SORTNO}} \ \begin{matrix}\underline{\text{EQUAL}}\\ \underline{=}\end{matrix} \ \underline{\text{sortie number}} \ \left[\underline{\&}\ \underline{\text{sortie number}}\ \underline{\&}\ . . .\right]$

9.3 Output

9.3.1 Standard Output

9.3.1.1 BUILD FILE SIDAC. Two tapes/files are generated; one for BLUE targets and the other for RED targets. There is no standard print.

9.3.1.2 BUILD FILE TABLE. The output consists of a tape/file on user direct unit. There is no standard output.

9.3.1.3 BUILD FILE OTHER. The output consists of a user defined output file. There is no standard print.

9.3.1.4 PLOTDATA. The output consists of a tape suitable for the CALCOMP plotter. There is no standard print.

9.3.1.5 PLOTIT. The output consists of a tape suitable for the CALCOMP plotter. On this plot a set of symbols is used to distinguish various sortie events. The list of these symbols appears in figure 24.

9.3.2 Optional Output. Print of records within files SIDAC, TABLE, and PLOTDATA can be obtained; table 14 gives the format of records within SIDAC; table 15 formats the TABLE records; figure 25 the PLOTDATA records, figure 26 the PLOTIT records, and figure 27 the error messages.

The optional output for BUILD FILE OTHER consists of a print of the retrieval scheme constructed during processing (see subsection 5.3.2.3).

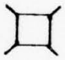
Event Type	Event	Acronym	Symbol
2	Launch bomber	LAUNB	☆
4	Refuel	LEREFUEL	$\overline{\underline{X}}$ (and zone number
5	Boundary or sector crossing	INSECTOR	Z (and zone number
8	Drop bomb	LOCLATTR	□ (number by sequential code)
14	Launch ASM	LAUNASM	X (and broken line to target)
15	Launch decoy	LAUNDCOY	D
16	Recovery	LANDHO	
13	Abort	LABORT	*
18	Go to high altitude	IGOHI	HI
19	Go to low altitude	IGOLOW	LO
20	Dogleg	LEGDOG	
10	Change time	ITIME	

Figure 24. PLOTIT Plot Symbols

Table 15. (Part 5 of 5)

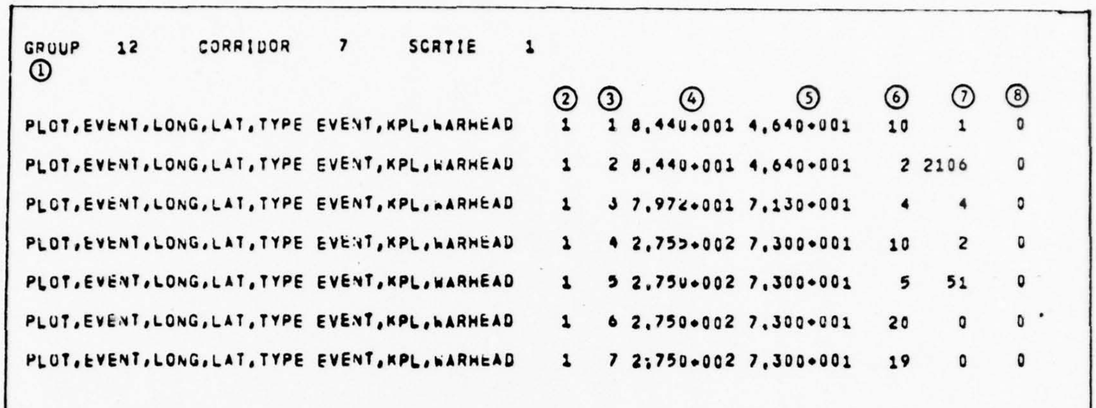
OFFENSIVE RECOVERY BASE LIST

Column	Meaning
1-7	'F1RECBS'
8	Blank
9	Side: 1 for Blue; 2 for Red
10-14	Line count
15	Blank
16-21	DESIG, alphabetic
22-23	Blank
24-30	Latitude (LAT), degrees, minutes, seconds
31-38	Longitude (LONG), degrees, minutes, seconds
39	Blank
40-45	NAME, alphabetic
46-49	World Area Code (WACNO), alphabetic
50-55	Bomber Encyclopedia Number (BENO), alphabetic
56-60	Category Code (CATCODE), numeric
61-62	Country Location (CNTRYL), alphabetic
63-68	Major Complex Number (MAJOR), numeric
69-70	TASK, alphabetic
71-75	Index Number (INDEXNO), numeric
76	Blank
77-90	Capacity (CAPACITY), numeric

```
①   MAP = PIC1  SIDE = BLUE  SCALE = 5000000.0
    PLOTS REQUESTED PENCOR DEPCOR REFUEL RECOV
    ORIGIN  40.00   250.00

②    PENETRATION CORRIDORS
    CORNUM = 1   ORLAT = 45.00  ORLONG = 267.00
         DOGLEG = 10  LAT = 45.00  LONG = 267.00
         DOGLEG = 20  LAT = 53.00  LONG = 270.00
    CORNUM = 2   ORLAT = 63.00  ORLONG = 310.00
         DOGLEG = 10  LAT = 63.00  LONG = 310.00
         DOGLEG = 20  LAT = 60.00  LONG = 315.00

③    DEPENETRATION CORRIDORS
    CORNUM = 1
         DOGLEG = 10  LAT = 75.00  LONG = 270.00
         DOGLEG = 20  LAT = 78.00  LONG = 265.00

④    REFUEL POINTS
    1    LAT = 83.00  LONG = 265.00
    2    LAT = 50.00  LONG = 310.00

⑤    RECOVERY BASES
    1    LAT = 45.00  LONG = 250.00
    CORNUM = 1 ORLAT = 75.00 ORLONG = 270.00

⑥   THERE WERE    0 POINTS OFF THE GRAPH

    THERE WERE   12 POINTS ON THE GRAPH
```

HEADING	MEANING
①	Print of input values including any defaults
②	Penetration corridor point, each corridor is followed by its dogleg
③	Depenetration corridor, corridor symbol is plotted at coordinates of first dogleg
④	Refuel point print including sequential count
⑤	Recovery base print. Corridor is a depenetration corridor and latitude and longitude are those of first dogleg
⑥	Termination message

Figure 25. PLOTDATA Optional Output



```
GROUP   12      CORRIDOR    7      SCRTIE    1
①
                                                ②  ③     ④            ⑤          ⑥     ⑦    ⑧
PLOT,EVENT,LONG,LAT,TYPE EVENT,KPL,WARHEAD      1   1 8,440+001 4,640+001    10     1    0
PLOT,EVENT,LONG,LAT,TYPE EVENT,KPL,WARHEAD      1   2 8,440+001 4,640+001     2  2106    0
PLOT,EVENT,LONG,LAT,TYPE EVENT,KPL,WARHEAD      1   3 7,972+001 7,130+001     4     4    0
PLOT,EVENT,LONG,LAT,TYPE EVENT,KPL,WARHEAD      1   4 2,755+002 7,300+001    10     2    0
PLOT,EVENT,LONG,LAT,TYPE EVENT,KPL,WARHEAD      1   5 2,750+002 7,300+001     5    51    0
PLOT,EVENT,LONG,LAT,TYPE EVENT,KPL,WARHEAD      1   6 2,750+002 7,300+001    20     0    0
PLOT,EVENT,LONG,LAT,TYPE EVENT,KPL,WARHEAD      1   7 2;750+002 7,300+001    19     0    0
```

Heading	Label	Description
①	-	Sortie identification
②	NUMPLOT	Plot number for this graph
③	KK	Event number for this sortie
④	SUBHLO	Longitude of this sortie event (degrees west)
⑤	SUBHLA	Latitude of this sortie event (degrees)
⑥	SUBJTP	Sortie event type (see table 5)
⑦	SUBKPL	Index number of launch bases, refuel areas, zone boundaries, targets, and recovery points; for launch decoy events it is the number of decoys launched
⑧	IWARHD	Warhead type index

Figure 26. PLOTIT Sortie Event Data

1 ERROR IN FILE CLAUSE

FILE clause contains error or is absent

2 ERROR IN WHERE CLAUSE

Attribute is not SIDE (TABLE only)

3 ERROR IN SETTINC CLAUSE

Illegal attribute or value (PLOTDATA only)

POINT (I3) OFF MAP X = (F10.3) Y-10 = (F10.3) - point coordinates exceed plot limits (PLOTDATA and PLOTIT only)

4 ERROR IN FORMAT CLAUSE

Probable cause is illegal special word (OTHER only)

5 ERROR IN DEFINE CLAUSE

DEFINE clause contains an error (OTHER only)

6 DEFINES CANNOT BE RESOLVED

No order can be found in which to properly execute define variable calculations (OTHER only)

7 ERROR IN SORT CLAUSE

Sort parameters illegal, missing or in the wrong order (OTHER only)

Figure 27. EIM Error Messages

APPENDIX A

QUICK ATTRIBUTE NAMES AND DESCRIPTIONS

This appendix lists, in alphabetical order, the attributes used in the CCTC QUICK data base. Also provided are the definition/description of each attribute as it pertains to the QUICK system.

There are three attribute groups: gaming, non-gaming, and text English inputs. Gaming attributes are those elements necessary to define targets, weapons, and geographic data. Non-gaming elements are used to assist in the definition of Integrated Data Base structure. Text English attributes are part of the user input command structure. In this appendix an asterisk (*) after the attribute name implies a non-gaming attribute; a pound sign (#) implies text English inputs; all other attributes are gaming related.

Attribute Name	Description
ACLASS*	Class to which type is assigned by Assignment Table
ACTIVE	Value which is set true if a target class is to be processed by the allocation subsystem
ADBLI	ALRTDB probability for initiative attack
ADBLR	ALRTDB probability for retaliatory attack
ADVERBVL*	Adverb's identifying number
ALPLSTVL*	Value in attribute edit check list
ALRTDB	Probability of destruction before launch (DBL) of alert delivery vehicle (missile or bomber)
ALRTDL	Delay of alert vehicle before commencing launch (hours)
ALTDLY	Alert delay
ARRIVE	Time of arrival of assigned weapon
ASGHOB	Height of Burst of assigned weapon
ASIDE	Attacking side
ASNTASK*	Task to which a range of DESIG values is assigned by Assignment Table
ATDEFALT*	Default value for attribute (in Directory)
ATRNGHI*	Upper limit for attribute (in Directory)
ATRNGLOW*	Lower limit for attribute (in Directory)
ATTRBTYP*	Value which describes an attribute as to mode-alpha, list, integer, float; and type-single, multiple, control
ATTRCD	Attrition rate prior to corridor
ATTRCO	Attrition parameter for penetration corridor
ATTRIBAD*	Attribute's common block address
ATTRIBN1*	Attribute's name (first half)
ATTRIBN2*	Attribute's name (second half)
ATTRIBNO*	Attribute's identifying number
ATTRLE	Attrition parameter for a penetration corridor dogleg
ATTRPRE1	Attrition parameter for precorridor dogleg 1
ATTRPRE2	Attrition parameter for precorridor dogleg 2
ATTRPRE3	Attrition parameter for precorridor dogleg 3
ATTRSU	Penetration corridor attrition parameter after defense suppression

Attribute Name	Description
ATYPE*	Legal type assignment in Assignment Table
BENO	Bomber Encyclopedia Number
BPENFAC	Multiplier for bomber attrition parameters
CAPACITY	Vehicle capacity for a recovery base
CATCODE	Target category code
CATHI*	Upper limit of category for type assignment
CATLO*	Lower limit of category for type assignment
CCREL	Command and control region reliability
CEP	Circular Error Probable, delivery error applicable to bomber and missile delivered weapons (nautical miles)
CEPASM	CEP for an air-to-surface missile
CHAINNAM*	Chain name in data structure index
CLASS	Class name assigned to identify various subsets of the data base
CLAUSESW*	Switch that identified verbs that require clauses
CLAUSETY*	Value that identifies an adverb's clause type (Boolean, Sequential, Single or Null)
CLOSE	Determines magnitude of closing force in allocation process
CLOSER	Controls rate of increase in closing force
CMISS	Constant used in missile time-of-flight calculation
CNFLG*	Flag specifying country owner or location restriction in Assignment Table
CNTRYLOC	Country code for country where target is located
CNTRYOWN	Country code for country which owns the target
CORBOMB	Number of nautical miles prior to corridor entry
CORMSL	Fraction of missile flight completed at time zero
CORNUM	Corridor identification number
CORR2	SMAT array multiplier for final allocation evaluation
CORR	SMAT array multiplier
COUNTRY*	Legal country codes in Assignment Table
DEFDIST1	Length of precorridor dogleg 1
DEFDIST2	Length of precorridor dogleg 2
DEFDIST3	Length of precorridor dogleg 3

Attribute Name	Description
DEFRAN	Typical range of interceptors at defense bases near a corridor (nautical miles)
DELTVAL	Maximum fractional difference in value allowed in allocators time-of-arrival cells
DESIG	Target designator code
DESIGA2*	Alpha portion of assignable DESIG in Assignment Table
DGZLAT	Offset distance from target latitude of weapon assignment
DGZLONG	Offset distance from target longitude of weapon assignment
DISPNAM1*	REPORT display name (first half)
DISPNAM2*	REPORT display name (second half)
DISTANCE	General distance attribute
DISTDF	Distance from target to end of depenetration corridor
DOGLEG	Corridor dogleg identification number
DSIDE	Defending side
ELEMNTTY*	For adverbs with element phrases -- type of element (e.g., operator, special word, etc.)
ELEMNTVL*	For ELEMNTTY of operator or special word, the exact operator or special word allowed
ERRCLOS	Controls termination of allocation processing
EXNBOMB	Number of vehicle 'loads' of weapons to be added to each bomber group
EXNMIRV	Same as EXNBOMB but for MIRVs
EXNMISS	Same as EXNBOMB but for single shot missiles
EXPASM	Fraction of bomber groups weapons which are ASMs
FACMIRV	Modifies SMAT array for MIRV systems
FFRAC	Fission fraction (fission yield/total yield)
FIXED	Switch to indicate if weapon assignment was user fixed
FIXOPT	Fixed assignment option
FLAG	Numeric code (1 through 9) used to impose allocation restrictions
FLMULT	Assignment range for weapon assignment to multiple targets
FSALVO	Salvo number of weapon assignment

Attribute Name	Description
FSNSTVTY	Controls sensitivity of multiplier adjustment during later part of allocation
FULL1*	Flag set when KOUNT1 is at its maximum
FULL2*	Flag set when KOUNT2 is at its maximum
FULL3*	Flag set when KOUNT3 is at its maximum
FULL4*	Flag set when KOUNT4 is at its maximum
FULL5*	Flag set when KOUNT5 is at its maximum
FUNCTI	Operational application code for a weapon system (e.g., ICBM)
FVALT1	Fraction of target value remaining at T1, T2, T3, T4, and T5, respectively
FVALT2	
FVALT3	
FVALT4	
FVALT5	
FVULN1	Fraction of value of target in first hardness component
GBASE	Number of launch bases in weapon group
GFRASM	Fraction of weapons in group which are ASMs
GLAT	Latitude of group centroid
GLONG	Longitude of group centroid
GNVEH	Number of vehicles in group
GNWPNADJ	Number of weapons in group to be allocated (includes any overallocation)
GNWPNS	Number of weapons in group
GPKNAV	Single shot kill probability of weapon group against a naval target
GREFCODE	Group refuel code
GREFTIME	Group refuel time
GROUP	Group identification number
GSBL	Probability of group's survival before launch
GSBLREAL	Same as GSBL but not adjusted for overallocation
GSTART	Starting weapon index
GTYPE	Group weapon type
GTYPREFC	Reference code of group's weapon type record
GYIELD	Group yield (megatons)

Attribute Name	Description
HAZ	Lethal radius for air burst for first hardness component
HAZ2	Same as HAZ but for second hardness component
HGZ	Same as HAZ but for ground burst
HGZ2	Same as HZ2 but for ground burst
HIGHFAC	Upper estimate of terminal ballistic missile capability
HILOAT	The ratio of the low altitude attrition rate to the high altitude attrition rate
IALERT	Alert status; 1 = alert, 2 = nonalert
ICLASS	Class index assigned for game
ICOMPL	Complex index
IDHOB	Preferred height of burst indicator
IGIW	Indices of General Industrial Worth (dollars)
IMATCH	Controls method used to determine if a weapon allocation has met the required minimum destruction fraction specified by MINKILL
INDEXNO	Index of a target used during processing to identify the target
INITSTRK	Side initiating attack
IPENMO	Penetration mode; 1 = aircraft uses penetration corridor, 0 = penetration corridor not used
IRECMO	Recovery mode; 1 = aircraft recovery planned, 0 = aircraft recovery not planned
IREFUEL	Bomber refueling code
IREG	Index to identify geographic region
IREP	Reprogramming index (capability of missile squadron)
ISITE	Site number
IVERIFY	Controls operation of allocator after final allocation
KORR	Corridor used by assigned weapon
KORSTY	Parameter to adjust mode of corridor penetration
KOUNT1*	Highest numeric value for a unique DESIGA2 in region 1
KOUNT2*	Highest numeric value for a unique DESIGA2 in region 2
KOUNT3*	Highest numeric value for a unique DESIGA2 in region 3

Attribute Name	Description
KOUNT4*	Highest numeric value for a unique DESIGA2 in region 4
KOUNT5*	Highest numeric value for a unique DESIGA2 in region 5
LABEL#	Input; plot label selection switch
LAT	Latitude (degrees)
LAW	Specifies form of damage law used for area targets
LBMBREF	Reference code of last bomber sortie record
LCHINT	Time between successive vehicle launches from the same base (missile or bomber) subject to the simultaneous launch condition
LINELENGTH#	Input; length of report print line
LMSLREF	Reference code of last missile sortie record
LONG	Longitude (degree)
LOWFAC	Lower estimate of terminal ballistic missile capability
MAJOR	Target major complex number
MAP#	Input; map type selection
MASDETNM*	Master or Detail Name in data structure index
MASDETNO*	Master or Detail Number in data structure index
MAXFRA	Maximum value of weapon resources to be used relative to target value
MAXKILL	Desired maximum damage expected for a target
MAXSAL	Maximum number of salvoes
MINCAP*	Lowest acceptable JAD capacity for type in Assignment Table
MINDAMAG	Minimum fraction of target value which must be destroyed by each weapon allocated
MINKILL	The required minimum damage established for a target
MINOR	Target minor compound number
MISDEF	Target ballistic missile defense
MYRECOV1	Depenetration corridor recovery base 1 (DESIG)
MYRECOV2	Depenetration corridor recovery base 2 (DESIG)
MYRECOV3	Depenetration corridor recovery base 3 (DESIG)
MYRECOV4	Depenetration corridor recovery base 4 (DESIG)
NADBLI	NLRTDB for initiative attack

Attribute Name	Description
NADBLR	NLRTDB for retaliatory attack
NALTDLY	Nonalert delay
NAME	Alphanumeric descriptor for any item in the data base
NAREADEC	Number of decoys per independent reentry vehicle for area BMD
NASMTYP	Number of distinct types of ASMs
NBASES	Total number of launch bases
NCCREG	Number of command and control regions
NCMS	Number of counter measures carried by vehicle
NCOMPLX	Number of target complexes
NDCNTYCD	Number of distinct country codes
NDECOYS	Number of decoys on a bomber or number of decoys per independent reentry vehicle for terminal BMD
NDEPCRD	Number of depenetration corridors
NFIXES	Number of fixed weapon assignments for group
NHRDCOMP	Number of hardness components
NLRTDB	Delay of nonalert vehicle before commencing launch (hours)
NLRTDL	Probability of destruction before launch (DBL) of nonalert vehicle
NMPSIT	Number of missiles per site
NOALER	Number of vehicles on alert status (also spelled NALERT)
NOINCO	Number of delivery vehicles on commission
NOPERSQ	Number of weapon vehicles per squadron
NPAYLOD	Number of payload tables
NPENCRD	Number of penetration corridors
NPNCRTY	Number of penetration corridor types
NPRCRDEF	Number of precorridor defense doglegs
NPRSQ1	Number per squadron - scenario 1
NPRSQ2	Number per squadron - scenario 2
NPRSQ3	Number per squadron - scenario 3
NPRSQ4	Number per squadron - scenario 4

Attribute Name	Description
NRECOVB	Number of recovery bases
NREFUEL	Number of refuel points
NSAL1 NSAL2 NSAL3	Number of weapons in salvo for salvoed weapon type (numbers are packed eight per word)
NSFIX1 NSFIX2 NSFIX3	Number of fixed assignments in salvo (numbers are packed eight per word)
NTANKERB	Number of tanker bases
NTARGETS	Number of targets as seen by the allocator
NTIMCOMP	Number of time components
NTINT	Number of terminal BMD interceptors at target
NUMALOC	Number of assignment by allocator
NUMDBL	Number of aircraft destroyed before launch
NUMLOAD	Number of warheads of a type in payload table
NWEPGRP	Number of weapon groups
NWEPTYP	Number of weapon types selected by user
NWHDS	Number of warheads per independent reentry vehicle (missile)
OFFLAT	Latitude of DGZ after offsets
OFFLONG	Longitude of DGZ after offset
ORLAT	Penetration corridor orientation point latitude
ORLONG	Penetration corridor orientation point longitude
PAGELENGTH#	Input; length of report print page (lines)
PAYALT	Bomber payload release altitude
PAYNAME	Payload table name (used in payload-weapon linking)
PAYTBLNM	Payload table name
PCTIW#	Scaling factor for IGIW calculations
PCTPOP#	Scaling factor for POP calculations
PDES	Probability that launch failure destroys missile
PDUD	Probability that warhead will fail to detonate
PEN	Penetration probability of assigned weapon
PEXBOMB	Fractional number of bomber weapons to be added by PLANSET
PEXMIRV	Same as PEXBOMB but for MIRVs

Attribute Name	Description
PEXMISS	Same as PEXBOMB but for single shot missiles
PFIW#	Scaling factor for IGIW calculations
PFPF	Probability of failure during powered flight
PFPOP	Scaling factor for POP calculations
PHRASETY*	Value that identifies an adverb's phrase type (relational, Equal or Like, Elemental)
PINC	Probability that a missile is in commission
PKNAV	Single shot kill probability of a weapon against a naval target (a value greater than zero restricts weapon use to naval targets)
PKTX	Probability of warhead kill by terminal BMD
PLABT	Probability of refuel abort
PRM	Controls value of quadratic premium
PROBHIGH	Probability that terminal defense is modified by HIGHFAC
PROBLOW	Probability that terminal defense is modified by LOWFAC
QUALITY	Controls extent to which STALL will attempt to refine allocation
RADIUS	Size descriptor for area targets (nautical miles)
RADPX	Probability of warhead kill by area BMD
RANGE	Vehicle range (nautical miles)
RANGEASM	Range of ASM
RANGEDEC	Range decrement for low altitude aircraft flight (high range/low range)
RANGEMOD#	Adjusted weapon group range
RANGEREF	Range of bomber with refueling
RATIOINT	Ratio of longest integration period used to theoretical
REGION*	Region assigned to country in Assignment table
REL	Reliability - probability that weapon system will arrive at target given successful launch
RELASM	ASM reliability
REPORTCODE*	Input; report code for REPORT module output
RETARGET#	Input parameter specifying missile may retarget

Attribute Name	Description
RINTPRD	Approximate ratio between rate of change of target weights between different integration periods
RNGMIN	Minimum range (nautical miles) for the missile type, used in computing flight times
RVAL	Relative value of weapon assignment
SCALE#	Input; map scale
SCENARIO#	DBMOD input scenario selection
SCHANGE	Alphanumeric indicator for a changed sortie
SCUMSURV	Cumulative survival probability for sortie event
SDAMEXP	Damage expected as a result of sortie event
SDELAY	Delay time for sortie
SDELTIME	Time change during sortie event
SDEPEN	Depenetration corridor used by sortie
SETTLE	Control the number of passes at PROGRESS = .75
SEVCODE	Sortie event code. Identified type of event
SIDE	Item side name
SIMLUN	Maximum number of vehicle launches which can occur simultaneously from one base
SLAT	Sortie launch base latitude
SLOCATTR	Local attrition rate for sortie event
SLONG	Sortie launch base longitude
SLOW	Available low distance for bomber sortie
SLOW1	
SLOW2	
SLOW3	
SNSTVTY	Controls sensitivity of multiplier adjustment during early phases
SORTNO	Sortie identifier code number
SPDLO	Speed at low altitude (knots)
SPEED	Speed (knots)
SPEEDASM	Speed of ASM (knots)
SPLACE	Used with SEVCODE as modifier
SREFUEL	IREFUEL code for sortie

Attribute Name	Description
STALADJ	Determines extent to which STALL favors high unit profit versus efficiency in selecting weapons
STARFAC	Multiplier level of bomber defense for sortie
SVEHNUM	Sortie vehicle number
T1 T2 T3 T4 T5	Times of departure of first through the fifth value components of a target
TABCHAR*	Dictionary tab character
TARDEF#	Level of target defense
TARDEFHI	Level of local bomber defenses at high altitude
TARDEFLO	Level of local bomber defenses at low altitide
TARFAC	Multiplier level of terminal bomber defenses
TASK	Target task code indicating targeting priority
TGTMULT	Target multiple number
TGTNUMB	Target index in target list as given to the allocator
TGTREFCD	Target IDS internal Reference Code (used in target list)
TINTFAC	Multiplier level of terminal BMD
TOFMIN	Minimum flight time (minutes) for missile types used in computing flight time
TTOS	Total time on station (for a tanker) (hours)
TYPE	Alphanumeric designator (type name) to identify sets in the data base
VAL	Relative value of an item within its CLASS as established in the data base by the user (also spelled VALUE)
VERBVAL*	Verb's identifying number
VONBASE	Number of launch bases plus index of starting vehicle
VOZ	Normalized target value
VULN1	First hardness component of a target
VULN2	Second hardness component of a target
WACNO	Target World Area Code Number
WEPNAME	Subset of weapon type
WHOB	Preset HOB for weapon

Attribute Name	Description
WORDSTR1*	First half of word in dictionary
WORDSTR2*	Second half of word in dictionary
WORDTYPE*	Identifies dictionary word as to type (i.e., Attribute, Verb, etc.)
WORDVAL*	Dictionary word identifying number within type
YIELD	Yield (Megatons)

APPENDIX B

QUICK DATA BASE DIRECTORY

The QUICK data base directory consists of a list of all the attributes which can be used to describe the data items defined in the integrated data base. The information contained in the directory for each attribute includes:

a. The name of the attribute plus an indicator that defines the attribute as belonging to a logical collection. If the name appears alone it is a gaming attribute; if an asterisk (*) follows the name it is a non-gaming attribute; if a pound sign (#) follows it is text English input

b. The type of the attribute may be:

   o Single - appears only within one record type. Input values are either LIST, INTGER, ALPHA, or FLOAT depending on the mode.

   o Multiple - appears within more than one record type. Input values are either MLTLST, MLTINT, MLTALP, or MLTFLT depending on the mode.

   o Control - same as multiple plus is used as a match key for internal structure definition. Input values are either CNTLST, CNTINT, CNTALP, or CNTFLT depending on the mode.

c. The modes, or input/output conversions. These are standard FORTRAN formats plus a list which specifies a list of alphabetic entries.

d. The default value to be assigned the attribute when it is not defined for an item.

e. The attribute lower limit

f. The attribute upper limit

Attribute Name	Type	Mode	Default	Lower Limit	Upper Limit
ACLASS*	Single	List	Other	-	-
ACTIVE	Single	Integer	0	0	0
ADBL1	Single	Float	o	o	1
ADBLR	Single	Float	o	o	1
ADVERBVL*	Control	Integer	0	0	0
ALPLSTVL*	Single	Alpha	-	-	-
ATTRCD	Single	Float	0	0	1
ATTRCO	Single	Float	0	0	1
ATTRIBAD*	Multiple	Integer	0	0	0
ATTRIBN1*	Control	Alpha	-	-	-
ATTRIBN2*	Control	Alpha	-	-	-
ATTRIBNO*	Multiple	Integer	0	0	322
ATTRLP	Single	Float	0	0	1
ATTRPRE1	Single	Float	0	0	1
ATTRPRE2	Single	Float	0	0	1
ATTRPRE3	Single	Float	0	0	1
ATTRSH	Single	Float	0	0	1
ATYPE*	Single	Alpha	-	-	-
BENO	Single	Alpha	-	-	-
BPENFAC	Single	Float	1	0	0
CAPACITY	Single	Integer	0	0	9999

Attribute Name	Type	Mode	Default	Lower Limit	Upper Limit
CATCODE	Single	Integer	75100	1	99999
CATHI*	Single	Integer	0	1	99999
CATLO*	Single	Integer	0	1	99999
CCREL	Single	Float	1	0	1
CEP	Single	Float	1	0	1
CEPASM	Single	Float	1	0	1
CHAINNAM*	Multiple	Alpha	--	--	--
CLASS	Multiple	Alpha	INDEX	--	--
CLAUSESW*	Single	Integer	0	0	1
CLAUSETY*	Single	Integer	2	1	4
CLOSE	Single	Float	1.05	0	0
CLOSER	Single	Float	4	0	0
CMISS	Single	Float	1	0	1
CNFLG*	Single	Integer	0	0	0
CNTRYLOC	Single	Alpha	--	--	--
CNTRYOWN	Single	Alpha	--	--	--
CORBOMB	Single	Float	0	0	0
CORMSL	Single	Float	0	0	0
CORNUM	Multiple	Integer	1	0	0
CORR	Single	Float	.5	0	0
CORR2	Single	Float	0	0	0
COUNTRY*	Single	Integer	--	--	--
DEFDIST1	Single	Float	0	0	0
DEFDIST2	Single	Float	0	0	0
DEFDIST3	Single	Float	0	0	0
DEFRAN	Single	Float	0	0	0
DELTVAL	Single	Float	.005	0	0
DESIG	Control	Alpha	ZZ999	--	--
DESIGA2	Multiple	Alpha	ZZ999	--	--
DGZLAT	Single	Float	0	-90	90
DGZLONG	Single	Float	0	0	360
DISPNAM1*	Single	Alpha	--	--	--

Attribute Name	Type	Mode	Default	Lower Limit	Upper Limit
DISPNAM2*	Single	Alpha	--	--	--
DISTANCE	Multiple	Float	0	0	0
DISTDF	Single	Float	0	0	0
DOGLEG	Multiple	Integer	1	0	0
DSIDE	Single	Alpha	--	--	--
ELEMNTTY*	Single	Integer	0	0	10
ELEMNTVL*	Single	Integer	0	0	999
ERRCLOS	Single	Float	.001	0	0
EXNBOMB	Single	Float	0	0	0
EXNMIRV	Single	Float	0	0	0
EXNMISS	Single	Float	0	0	0
EXPASM	Single	Float	0	0	0
FACMIRV	Single	Float	0	0	0
FFRAC	Single	Float	1	0	1
FIXED	Single	Integer	0	0	0
FIXOPT	Single	List	TRUE	--	--
FLAG	Multiple	Integer	0	0	9
FLMULT	Single	Integer	0	0	0
FSALVO	Single	Integer	1	1	24
FSNSTVTY	Single	Float	1	0	0
FULL1*	Single	Integer	0	0	0
FULL2*	Single	Integer	0	0	0
FULL3*	Single	Integer	0	0	0
FULL4*	Single	Integer	0	0	0
FULL5*	Single	Integer	0	0	0
FUNCTI	Single	List	--	--	--
FVALT1	Multiple	Float	1	0	1
FVALT2	Multiple	Float	1	0	1
FVALT3	Multiple	Float	1	0	1
FVALT4	Multiple	Float	1	0	1
FVALT5	Multiple	Float	1	0	1
FVULN1	Multiple	Float	1	0	1

Attribute Name	Type	Mode	Default	Lower Limit	Upper Limit
GBASE	Single	Integer	0	0	0
GFRASM	Single	Float	0	0	1
GLAT	Single	Float	0	-90	90
GLONG	Single	Float	0	0	360
GNVEH	Single	Integer	1	1	999
GNWPNADJ	Single	Integer	1	1	999
GNWPNS	Single	Integer	1	1	999
GPKNAV	Single	Float	0	0	0
GREFCODE	Single	Integer	1	0	1
GREFTIME	Single	Float	100	0	0
GROUP	Control	Integer	0	0	999
GSBL	Single	Float	0	0	0
GSBLREAL	Single	Float	0	0	0
GSTART	Single	Integer	0	0	0
GTYPE	Single	Alpha	--	--	--
GTYPREFC	Single	Integer	0	0	0
GYIELD	Single	Float	0	0	99
HAZ	Multiple	Float	0	0	0
HAZ2	Multiple	Float	0	0	0
HGZ	Multiple	Float	0	0	0
HGZ2	Multiple	Float	0	0	0
HIGHFAC	Single	Float	0	0	0
HILOAT	Single	Float	.1	0	1
IALERT	Single	Integer	0	0	0
ICLASS	Single	Integer	1	1	15
ICOMPL	Control	Integer	0	1	999
IDHOB	Multiple	Integer	0	0	0
IGIW	Single	Integer	0	0	99999
IMATCH	Single	Integer	0	0	0
INDEXNO	Multiple	Integer	1	1	99999
INITSTRK	Single	Integer	2	1	2
IPENMO	Single	Integer	1	0	1

Attribute Name	Type	Mode	Default	Lower Limit	Upper Limit
IREFUEL	Single	Integer	0	0	20
IREG	Control	Integer	0	0	20
IREP	Single	Integer	0	0	5
ISITE	Single	Integer	0	0	100
IVERIFY	Single	Integer	0	0	2
KORR	Single	Integer	0	1	999
KORSTY	Single	Integer	0	0	10
KOUNT1*	Single	Integer	0	0	999
KOUNT2*	Single	Integer	0	0	999
KOUNT3*	Single	Integer	0	0	999
KOUNT4*	Single	Integer	0	0	999
KOUNT5*	Single	Integer	0	0	999
LABEL#	Single	Alpha	--	--	--
LAT	Multiple	Float	0	-90	90
LAW	Single	List	POWER	--	--
LBMBREF	Single	Integer	0	0	0
LCHINT	Single	Float	0	0	999
LWELENGTH#	Single	Integer	120	0	0
LMSLREF	Single	Integer	0	0	0
LONG	Multiple	Float	0	0	360
LOWFAC	Single	Float	0	0	0
MAJOR	Single	Integer	0	1	999999
MAP#	Single	Alpha	--	--	--
MASDETNM*	Multiple	Alpha	--	--	--
MASDETNO*	Multiple	Integer	0	0	0
MAXFRA	Multiple	Float	10	0	10
MAXKILL	Multiple	Float	1	0	1
MAXSAL	Single	Integer	0	0	0
MINCAP*	Single	Integer	0	0	0
MINDAMAG	Single	Float	0	0	0
MINKILL	Multiple	Float	0	0	1
MINOR	Single	Integer	0	0	999999

Attribute Name	Type	Mode	Default	Lower Limit	Upper Limit
MISDEF	Multiple	Integer	0	0	0
MYRECOV1	Single	Alpha	--	--	--
MYRECOV2	Single	Alpha	--	--	--
MYRECOV3	Single	Alpha	--	--	--
MYRECOV4	Single	Alpha	--	--	--
NADBLI	Single	Float	0	0	1
NADBLR	Single	Float	0	0	1
NALERT	Single	Integer	0	0	0
NALTDLY	Single	Float	0	0	0
NAME	Multiple	Alpha	--	--	--
NAREADEC	Single	Integer	0	0	50
NASMTYP	Single	Integer	0	0	999
NBASES	Single	Integer	0	0	999
NCCREG	Single	Integer	0	0	9
NCMS	Single	Integer	0	0	5
NCOMPLX	Single	Integer	0	0	999
NDCNTYCD	Single	Integer	0	0	9
NDECOYS	Single	Integer	0	0	10
NDEPCRD	Single	Integer	0	0	9
NFIXES	Single	Integer	0	0	30
NHRDCOMP	Multiple	Integer	1	1	2
NLRTDB	Single	Float	0	0	0
NLRTDL	Single	Float	0	0	0
NMPSIT	Single	Integer	0	0	999
NOINCO	Single	Integer	0	0	999
NOPERSQ	Single	Integer	0	0	999
NPAYLOD	Single	Integer	0	0	999
NPENCRD	Single	Integer	0	0	999
NPNCRTY	Single	Integer	0	0	999
NPRCRDEF	Single	Integer	0	0	3
NPRSQ1	Single	Integer	0	0	999
NPRSQ2	Single	Integer	0	0	999

Attribute Name	Type	Mode	Default	Lower Limit	Upper Limit
NPRSQ3	Single	Integer	0	0	999
NPRSQ4	Single	Integer	0	0	999
NRECOVB	Single	Integer	0	0	999
NREFUEL	Single	Integer	0	0	999
NSAL1	Single	Integer	0	0	0
NSAL2	Single	Integer	0	0	0
NSAL3	Single	Integer	0	0	0
NSFIX1	Single	Integer	0	0	0
NSFIX2	Single	Integer	0	0	0
NSFIX3	Single	Integer	0	0	0
NTANKERB	Single	Integer	0	0	999
NTARGETS	Single	Integer	0	0	999
NTIMCOMP	Multiple	Integer	0	0	999
NTINT	Multiple	Integer	0	0	999
NUMALOC	Single	Integer	0	0	999
NUMDBL	Single	Integer	0	0	99
NUMLOAD	Single	Integer	0	0	99
NWEPGRP	Single	Integer	0	0	999
NWEPTYP	Single	Integer	0	0	999
NWHDS	Single	Integer	1	1	16
OFFLAT	Single	Float	0	-90	90
OFFLONG	Single	Float	0	0	360
ORLAT	Single	Float	0	-90	90
ORLONG	Single	Float	0	0	360
PAGELENGTH[#]	Single	Integer	55	0	0
PAYALT	Single	List	--	--	--
PAYNAME	Single	Alpha	--	--	--
PAYTBLNM	Control	Alpha	--	--	--
PCTIW[#]	Single	Float	0	0	0
PCTPOP[#]	Single	Float	0	0	1
PDES	Single	Float	0	0	1
PDUD	Single	Float	0	0	0
PEN	Single	Float	1	0	1

Attribute Name	Type	Mode	Default	Lower Limit	Upper Limit
PEXBOMB	Single	Float	0	0	0
PEXMIRV	Single	Float	0	0	0
PEXMISS	Single	Float	0	0	0
PFIW#	Single	Float	0	0	0
PFPF	Single	Float	0	0	1
PFPOP	Single	Float	0	0	0
PHRASETY*	Single	Integer	1	1	3
PINC	Single	Float	0	0	1
PKNAV	Single	Float	0	0	1
PKTX	Single	Float	0	0	0
PLABT	Single	Float	0	0	1
PLOT#	Single	Alpha	--	--	--
POP	Single	Float	0	0	30000
PRABT	Single	Float	0	0	1
PRM	Single	Float	.5	0	0
PROBHIGH	Single	Float	0	0	0
PROBLOW	Single	Float	0	0	0
QUALITY	Single	Float	.5	0	0
RADIUS	Multiple	Float	0	0	50
RADPX	Single	Float	0	0	0
RANGE	Single	Float	0	0	20000
RANGEASM	Single	Float	0	0	20000
RANGEDEC	Single	Float	1.0001	0	10
RANGEMOD#	Single	Float	0	0	0
RANGEREF	Single	Float	0	0	20000
RATIOINT	Single	Float	2	0	0
REGION*	Single	Integer	1	1	4
REL	Single	Float	1	0	1
RELASM	Single	Float	1	0	1
REPORTCODE#	Single	Integer	42	0	0
RETARGET#		Alpha	--	--	--
RINTPRD	Single	Float	2	0	0

Attribute Name	Type	Mode	Default	Lower Limit	Upper Limit
RNGMIN	Single	Float	0	0	20000
RVAL	Single	Float	0	0	0
SCALE#	Single	Float	0	0	0
SCENARIO#	Single	List	INDIA	--	--
SCHANGE	Multiple	Alpha	--	--	--
SCUMSURV	Multiple	Float	0	0	0
SDAMEXP	Multiple	Float	0	0	0
SDELAY	Single	Float	0	0	0
SDELTIME	Multiple	Float	0	0	0
SDEPEN	Single	Integer	0	0	30
SETTLE	Single	Float	1	0	0
SEVCODE	Multiple	Integer	0	0	99
SIDE	Multiple	List	RED	--	--
SIMLUN	Single	Integer	1	1	50
SLAT	Single	Float	0	-90	90
SLOCATTR	Multiple	Float	0	0	1
SLONG	Single	Float	0	0	360
SLOW	Single	Float	0	0	0
SLOW1	Single	Float	0	0	0
SLOW2	Single	Float	0	0	0
SLOW3	Single	Float	0	0	0
SNSTVTY	Single	Float	.1	0	0
SORTNO	Single	Integer	0	0	999
SPDLO	Single	Float	0	0	10000
SPEED	Single	Float	0	0	30000
SPEEDASM	Single	Float	0	0	3000
SPLACE	Multiple	Integer	0	0	99
SREFUEL	Single	Integer	0	0	99
STALADJ	Single	Float	.5	0	0
STARFAC	Single	Float	0	0	0
SVEHNUM	Single	Integer	0	0	99
T1	Multiple	Float	1000	0	1000

Attribute Name	Type	Mode	Default	Lower Limit	Upper Limit
T2	Multiple	Float	1000	0	1000
T3	Multiple	Float	1000	0	1000
T4	Multiple	Float	1000	0	1000
T5	Multiple	Float	1000	0	1000
TABCHAR*	Single	Alpha	--	--	--
TARDEF#	Single	List	--	--	--
TARDEFHI	Multiple	Integer	0	0	7
TARDEFLO	Multiple	Integer	0	0	7
TARFAC	Single	Float	.1	0	0
TASK	Multiple	Alpha	--	--	--
TGTMULT	Single	Integer	0	0	0
TGTNUMB	Multiple	Integer	0	0	0
TGTREFCD	Single	Integer	0	0	0
TINTFAC	Single	Float	1	0	0
TOFMIN	Single	Float	0	0	1000
TTOS	Single	Float	0	0	100
TYPE	Multiple	Alpha	--	--	--
VALUE	Multiple	Float	.01	0	99999
VERBVAL*	Control	Integer	0	0	0
VONBASE	Single	Integer	0	0	0
VOZ	Multiple	Float	0	0	0
VULN1	Single	Alpha	--	--	--
VULN2	Single	Alpha	--	--	--
WACNO	Single	Alpha	--	--	--
WEPNAME	Single	Alpha	--	--	--
WHOB	Single	Alpha	--	--	--
WORDSTR1	Single	Alpha	--	--	--
WORDSTR2	Single	Alpha	--	--	--
WORDTYPE	Single	Integer	0	0	11
WORDVAL	Single	Integer	0	0	0
YIELD	Single	Float	0	0	9999

ATTRIBUTE NUMBER	ATTRIBUTE NAME	ATTRIBUTE COMMON BLOCK ADDRESS
91	ICOMPL	109
92	LAT	110
93	LONG	111
94	HGZ	112
95	HGZ2	113
96	HAZ	114
97	HAZ2	115
98	MAXKILL	116
99	MINKILL	117
100	MAXFRA	118
101	INDEXNO	119
102	RADIUS	120
103	TASK	121
104	VAL	122
105	VOZ	123
106	IDHOB	124
107	TARDEFHI	125
108	TARDEFLO	126
109	MISDEF	127
110	NTINT	128
111	NAME	129
112	CCREL	133
113	PAYTBLNM	135
114	YIELD	137
115	FFRAC	138
116	PDUD	139
117	SPEEDASM	140
118	NWHDS	140
119	RELASM	141
120	RANGEASM	142

ATTRIBUTE NUMBER	ATTRIBUTE NAME	ATTRIBUTE COMMON BLOCK ADDRESS
121	CEPASM	143
122	CEP	145
123	SPEED	146
124	NMPSIT	147
125	RANGE	148
126	ALTDLY	149
127	NALTDLY	150
128	FUNCTI	151
129	LCHINT	152
130	SIMLUN	153
131	MAXSAL	154
132	ACTIVE	155
133	IREP	157
134	PRABT	158
135	PLABT	159
136	RNGMIN	160
137	TOFMIN	161
138	CMISS	162
139	PDES	163
140	PFPF	164
141	SPDLO	157
142	RANGEDEC	158
143	RANGEREF	159
144	REL	156
145	IRECMO	160
146	IPENMO	161
147	NOINCO	188
148	TTOS	157
149	NUMALOC	58
150	STARFAC	166

ATTRIBUTE NUMBER	ATTRIBUTE NAME	ATTRIBUTE COMMON BLOCK ADDRESS
151	CATCODE	169
152	MAJOR	170
153	MINOR	171
154	WACNO	172
155	BENO	173
156	IGIW	174
157	ISITE	175
158	POP	176
159	DOGLEG	177
160	ATTRLE	178
161	DISTANCE	179
162	NUMLOAD	183
163	NUMASSGN	185
164	ARRIVE	186
165	NOPERSQ	187
166	GSTART	92
167	VONBASE	189
168	NOALER	190
169	ADBLI	191
170	ADBLR	192
171	ALRTDB	193
172	ALRTDL	194
173	NADBLI	195
174	NADBLR	196
175	NLRTDB	197
176	NLRTDL	198
177	PKNAV	199
178	IREFUEL	200
179	NPRSQ1	201
180	NPRSQ2	202

ATTRIBUTE NUMBER	ATTRIBUTE NAME	ATTRIBUTE COMMON BLOCK ADDRESS
181	NPRSQ3	203
182	NPRSQ4	204
183	NUMDBL	205
184	CAPACITY	207
185	MYRECOV1	105
186	MYRECOV2	106
187	MYRECOV3	107
188	MYRECOV4	108
189	TGTNUMB	209
190	TGTREFCD	210
191	ASIDE	211
192	NASMTYP	213
193	NBASES	214
194	NCCREG	215
195	NDCNTYCD	216
196	NDEPCRD	217
197	NPAYLOD	218
198	NPENCRD	219
199	NPNCRTY	220
200	NRECOVB	221
201	NREFUEL	222
202	NTANKERB	223
203	NTARGETS	224
204	NCOMPLX	225
205	NWEPGRP	226
206	NWEPTYP	227
207	CORBOMB #	229
208	CORMSL #	230
209	EXNBOMB #	234
210	EXNMIRV #	235

ATTRIBUTE NUMBER	ATTRIBUTE NAME	ATTRIBUTE COMMON BLOCK ADDRESS
211	EXNMISS	236
212	INITSTRK	228
213	MAP#	0
214	PCTIW#	0
215	PCTPOP#	0
216	PEXBOMB#	231
217	PEXMIRV#	232
218	PEXMISS#	233
219	PFIW#	0
220	PFPOP#	0
221	PLOT#	0
222	SCALE#	0
223	SCENARIO#	0
224	LABEL#	0
225	RANGEMOD#	0
226	TARDEF#	0
227	PAYALT	137
228	NDECOYS	138
229	NAREADEC	139
230	NCMS	140
231	RETARGET#	0
232	GBASE	93
233	GPKNAV	94
234	DEFRAN	104
235	PINC	165
236	PAGELENGTH#	0
237	LINELENGTH#	0
238	REPORTCODE#	0
239	DSIDE	212
240	WHOB	154
241	TGTMULT	131

ATTRIBUTE NUMBER	ATTRIBUTE NAME	ATTRIBUTE COMMON BLOCK ADDRESS
242	KORR	184
243	RVAL	237
244	PEN	238
245	DGZLAT	239
246	DGZLONG	240
247	OFFLAT	241
248	OFFLONG	242
249	FLMULT	243
250	GTYPREFC	78
251	GNWPNADJ	86
252	EXPASM	244
253	NFIXES	246
254	NSAL1	247
255	NSAL2	248
256	NSAL3	249
257	NSFIX1	250
258	NSFIX2	251
259	NSFIX3	252
260	NPRCRDEF	253
261	DEFDIST1	254
262	DEFDIST2	255
263	DEFDIST3	256
264	ATTRPRE1	257
265	ATTRPRE2	258
266	ATTRPRE3	259
267	ATTRCD	180
268	DISTDF	181
269	PAYNAME	135
270	BPENFAC	260

ATTRIBUTE NUMBER	ATTRIBUTE NAME	ATTRIBUTE COMMON BLOCK ADDRESS
271	CLOSE	261
272	CLOSER	262
273	CORR	263
274	CORR2	264
275	DELTVAL	265
276	ERRCLOS	266
277	FACNURV	267
278	FIXOPT	268
279	FSNSTVTY	269
280	HIGHFAC	270
281	IMATCH	271
282	IVERIFY	272
283	LAW	273
284	LOWFAC	274
285	MINDAMAG	275
286	PKTX	276
287	PRM	277
288	PROBHIGH	278
289	PROBLOW	279
290	QUALITY	280
291	RADPX	281
292	RATIOINT	282
293	RINTPRD	283
294	SETTLE	284
295	SNSTVTY	285
296	STALADJ	286
297	TARFAC	287
298	TINTFAC	288
299	FIXED	183
300	ASGHOB	208

ATTRIBUTE NUMBER	ATTRIBUTE NAME	ATTRIBUTE COMMON BLOCK ADDRESS
301	FSALVO	185
302	WEPNAME	135
303	SLAT	290
304	SLONG	291
305	SREFUEL	292
306	SDELAY	293
307	SDEPEN	294
308	SVEHNUM	295
309	SLOW	296
310	SLOW1	297
311	SLOW2	298
312	SLOW3	299
313	SLOCATTR	300
314	SCUMSGRV	301
315	SDELTIME	302
316	SDAMEXP	303
317	SEVCODE	304
318	SPLACE	305
319	SCHANGE	306
320	LBMBREF	167
321	LMSLREF	168
322	SORTNO	289

APPENDIX D

QUICKS NON-ATTRIBUTE WORD IDENTIFICATION NUMBER

This appendix contains all dictionary non-attribute words plus their associated internal identification number and the grammatical group each word belongs to.

DICTIONARY WORD	IDENTIFICATION NUMBER	GRAMMATICAL GROUP
","	1	Operator
"("	2	Operator
")"	3	Operator
AND "&"	4	Operator
OR	5	Operator
NOT	6	Operator
EQUAL "="	7	Operator
EQUALS "="	7	Operator
GREATER ">"	8	Operator
LESS "<"	9	Operator
BETWEEN	10	Operator
LIKE	11	Operator
PLUS "+"	12	Operator
MINUS "-"	13	Operator
TIMES "*"	14	Operator
DIVIDED "/"	15	Operator
POWER "**"	16	Operator
OF	17	Operator
ALTER	1	Verb
ASSIGN	2	Verb
BUILD	3	Verb
CHANGE	4	Verb
CREATE	5	Verb
DELETE	6	Verb
DESIGN	7	Verb
EDIT	8	Verb
INDEX	9	Verb
MODIFY	10	Verb
PLANSET	11	Verb
PLOTDATA	12	Verb
PREPARE	13	Verb
PRINT	14	Verb
RESTORE	15	Verb
SAVE	16	Verb

DICTIONARY WORD	IDENTIFICATION NUMBER	GRAMMATICAL GROUP
SELECT	17	Verb
ASTERISK	18	Verb
ALLOCATE	19	Verb
DGZSELECT	20	Verb
EVALUATE	21	Verb
FOOTPRINT	22	Verb
PLANOUT	23	Verb
PLOTIT	24	Verb
POSTALOC	25	Verb
ALPHAS	1	Adverb
RECALC	2	Adverb
ATTACKERS	3	Adverb
DEFENDERS	4	Adverb
DEFINE	5	Adverb
DISPLAY	6	Adverb
FIELDS	7	Adverb
FILE	8	Adverb
FIX	9	Adverb
FORMAT	10	Adverb
ONPRINTS	11	Adverb
OMITTING	12	Adverb
PLAYERS	13	Adverb
PRIORITY	14	Adverb
REPLACING	15	Adverb
SAME	16	Adverb
SETTING	17	Adverb
SORT	18	Adverb
SUPRESSING	19	Adverb
UNIT	20	Adverb
USING	21	Adverb
WHERE	22	Adverb
WITH	23	Adverb

DICTIONARY WORD	IDENTIFICATION NUMBER	GRAMMATICAL GROUP
VNOPTION	24	Adverb
KEEPING	25	Adverb
ORDER	26	Adverb
UICLASSES	27	Adverb
FLAGREST	28	Adverb
LOCREST	29	Adverb
MINRANGE	30	Adverb
MIRVREST	31	Adverb
PUNCH	32	Adverb
RANGEMOD	33	Adverb
READMUL	34	Adverb
SMAT	35	Adverb
ABTAPE	36	Adverb
ACARD	37	Adverb
CCARD	38	Adverb
COUNTRIES	39	Adverb
FINDMIN	40	Adverb
EQUATE	41	Adverb
FUNCOM	42	Adverb
GAMETIME	43	Adverb
ICARD	44	Adverb
IF	45	Adverb
PLANE	46	Adverb
REEQUATE	47	Adverb
STRIKE	48	Adverb
TGTMOD	49	Adverb
WPNMOD	50	Adverb
MISTME	51	Adverb
MSLCOR	52	Adverb
A	1	Special Word
ASCENDING	1	Special Word
AFTER	2	Special Word
D	3	Special Word

DICTIONARY WORD	IDENTIFICATION NUMBER	GRAMMATICAL GROUP
DESCENDING	3	Special Word
HEADER	4	Special Word
IN	5	Special Word
LINE	6	Special Word
TABLE	7	Special Word
NEW	8	Special Word
OLD	9	Special Word
OTHER	10	Special Word
PAGE	11	Special Word
REMOVE	12	Special Word
REPLACE	13	Special Word
SIDAC	14	Special Word
SPACES	15	Special Word
X	15	Special Word
TRAILER	16	Special Word
OWNED	18	Special Word
PAGENO	19	Special Word
ACOS	20	Special Word
ACOT	21	Special Word
ASIN	22	Special Word
ATAN	23	Special Word
AZIMUTH	24	Special Word
C	25	Special Word
COS	26	Special Word
COT	27	Special Word
EXP	28	Special Word
G	29	Special Word
HHR	30	Special Word
KDAY	31	Special Word
KMON	32	Special Word
KYEAR	33	Special Word
N	34	Special Word
SIN	35	Special Word

DICTIONARY WORD	IDENTIFICATION NUMBER	GRAMMATICAL GROUP
UPDATE	36	Special Word
INCLUDE	37	Special Word
EXCLUDE	38	Special Word
TAN	39	Special Word

Table 20. Warhead Attributes

BOMB, ASM, RV, MRV, and MIRV	
ATTRIBUTE NAME	COMMENT
FFRAC	Set by user
PDUD	Set by user
SIDE	Set by user
TYPE	Set by user
YIELD	Set by user
ASM	
CEPASM	Set by user
RANGEASM	Set by user
RELASM	Set by user
SPEEDASM	Set by user
MIRV	
NWHDS	Set by user
FACTOR	
NAREADEC	Set by user
NCMS	Set by user
NDECOYS*	Set by user
PAYALT	Set by user
TYPE	Set by user

*The range of decoys at low and high altitude is data set to 200 and 400 nautical miles respectively.

A payload table contains its name (attribute PAYTBLNM), the type(s) (attribute TYPE) of warhead(s) within the table and the number of times each warhead appears in the payload table (attribute NUMLOAD) as well as a side definition (attribute SIDE).

Each new weapon type created by JLM has a weapon subtype attached to it with the WEPNAME attribute containing a value equal to the TYPE attribute. This subtype is in turn linked to a payload table where the PAYTBLNM attribute is assigned a value equal to the TYPE attribute. However, the user needs to create new payload tables for those cases where a given type has more than one payload. These extra payload tables are created by specifying values for attribute PAYTBLNM. An example is:

```
CREATE SETTING CLASS=PAYLOD PAYTBLNM='B-47A'&'B-47B'&'B-47C'&
               B52GH1&B52GH2&'B-52E1'&'B-52E2'&'B-58'&
               'MM-II'&'MM-IB'&'MM-IA'&'TITAN'&
               POSEID&'POL-A2'&'POL-A3' SIDE=BLUE
```

Now the user links each payload table to its warheads. Examples are:

```
  CREATE SETTING CLASS=BOMB (PAYTBLNM,TYPE,NUMLOAD)=
         ('B-47A','MK-5',2)&('B-47B','MK-7',2)&
         ('B-47C','MK-18',1)&(B52GH1,'MK-5',4)&
         (B52GH2,'MK-18',2)&('B-52E1','MK-7',2)&
         ('B-52E2','MK-7',2)&('B-58','MK-18',1)&
         ('B-58','MK-5',3) SIDE=BLUE
  CREATE SETTING CLASS=ASM TYPE=HNDDOG NUMLOAD=2
                 SIDE=BLUE PAYTBLNM=B52GH1&B52GH2&'B-52E2'
         SETTING CLASS=RV SIDE=BLUE (PAYTBLNM,TYPE,NUMLOAD)=
                 ('MM-IB','MK-5',1)&('MM-IA','MK-5',1)&
                 (TITAN,'MK-18',1)
```

Weapon Base to Payload Link. JLM links all weapon bases to the weapon subtype with the same WEPNAME as the type name (TYPE). For cases where the user wishes to have more than one payload table for a type he must create a new weapon subtype and then relink the appropriate bases to the new weapon subtype. The first step is to create new subtypes. The connection is through attribute WEPNAME which is first linked to payloads and then defined for individual bases. To properly identify the appropriate payload tables for this process, the attribute PAYNAME is used instead of PAYTBLNM. Commands to create the new subtype, properly linked to the appropriate payload tables could be:

```
CREATE SETTING CLASS=BMBWEP SIDE=BLUE
                (WEPNAME,TYPE,PAYNAME)=
            ('B-52H1','B-52H' ,B52GH1 ) &
            ('B-52H2','B-52H' ,B52GH2 ) &
            ('B-52G1','B-52G' ,B52GH1 ) &
            ('B-52G2','B-52G' ,B52GH2 ) &
            ('B-52E1','B-52E' ,'B-52E1') &
            ('B-52E2','B-52E' ,'B-52E2') &
            ('B-58A' ,'B-58'  ,'B-58'  ) &
            ('F-111A','F-111' ,'B-47A' ) &
            ('F111BA','F-111B','B-47A')
CREATE SETTING CLASS=MSLWEP SIDE=BLUE
            (WEPNAME,TYPE, PAYNAME)=
            ('MM-IAA','MM-IA','MM-IA') &
            ('MM-IBA','MM-IB','MM-IB') &
            (MMIIIA,'MM-III','MM-II') &
            ('MM-IIA','MM-II','MM-II') &
            (TITANA,TITAN,TITAN) &
            (POLA2A,'POL-A2','POL-A2') &
            (POLA3A,'POL-A3','POL-A3') &
            (POSIDA,POSEID,POSEID)
```

The final step of weapon linkage is to relink the weapon bases to the new weapon subtype (WEPNAME). Relinkage is accomplished through a CHANGE verb and a WHERE clause used to identify the subset of bases in question and the SETTING clause used to name the subtype (WEPNAME) into which these bases fall. Examples are:

```
CHANGE WHERE CLASS=BOMBER AND SIDE=BLUE
       AND CATCODE BETWEEN 95 AND 98
       SETTING WEPNAME='B-52G2'
CHANGE WHERE CLASS=BOMBER AND SIDE=BLUE AND TYPE='B-52E'
       SETTING WEPNAME='B-52E1'
CHANGE WHERE CLASS=BOMBER AND SIDE=BLUE
       AND CATCODE BETWEEN 103 AND 111
       SETTING WEPNAME='B-52E2'
```

Geographic Classes. Penetration/depenetration corridor (CLASS=PENCOR or DEPCOR) and refuel points (CLASS=REFUEL) may be created at any stage prior to the execution of the PREPALOC module. Required attributes are given in table 21.

When creating corridors, the user should create the corridor itself first and in a separate verb, create the doglegs of the corridor. An example of a command that creates many penetration corridors is:

Table 21. Geographic Attributes

PENCOR

ATTRIBUTE NAME	COMMENT
ATTRCO	Set by user
ATTRLE	Set by user (dogleg attribute)
ATTRSU	Set by user
CORNUM	Set by user (corridor number)
DEFRAN	Set by user
DOGLEG	Set by user (dogleg attribute)
HILOAT	Set by user
KORSTY	Set by user
LAT	Set by user (dogleg attribute)
LONG	Set by user (dogleg attribute)
ORLAT	Set by user
ORLONG	Set by user

DEPCOR

CORNUM	Set by user (corridor number)
DOGLEG	Set by user (dogleg attribute)
LAT	Set by user (dogleg attribute)
LONG	Set by user (dogleg attribute)
MYRECOV1	Set by user (base DESIG)
MYRECOV2	Set by user (base DESIG)
MYRECOV3	Set by user (base DESIG)
MYRECOV4	Set by user (base DESIG)

REFUEL

IREG	Set by user
LAT	Set by user
LONG	Set by user

```
CREATE SETTING CLASS=PENCOR SIDE=BLUE
       (CORNUM,ORLAT,ORLONG,KORSTY,HILOAT,DEFRAN,ATTRSU,ATTRCO)=
       (  1, 50,350,1, .2,250,.0003,  .001) &
       (  2, 50,350,1. .2,250,.0003,  .001) &
       (  3, 55,215,4. .2,250,.0001, .0003) &
       (  4, 60,230,4, .2,250,.0001, .0003) &
       (  5, 50,235,4, .2,250,.0001, .0003) &
       (  6, 40,243,4, .2,250,.0001, .0003) &
       (  7, 60,275,2. .2,250,0.    , .0006)
```

For each corridor, doglegs start at the corridor origin and work outward. A line segment is identified through attribute DOGLEG. Each corridor will store line segments <u>sorted</u> on the attribute DOGLEG. Any number is permitted for DOGLEG entries. It is suggested that DOGLEG values be initially entered in multiples of 10 (10, 20, 30, etc.). This numbering will permit ease of redefinition if added line segments are defined after initial creation. A command that links DOGLEGs to corridors is:

```
CREATE SETTING CLASS=PENCOR SIDE=BLUE DOGLEG=10
       (CORNUM,LAT,LONG,ATTRLE)=
       ( 3,62,172,0.    ) & ( 4,68,199,0.    ) &
       ( 5,40,227,0.    ) & ( 6,36,236,0.    ) &
       ( 7,73,275,0.    ) & ( 8,52.5,349,.0005) &
       ( 9,43,342,0.    ) & (10,43,325,0.    ) &
       (11,71,330,0.    ) & (12,60,340,0.    )
```

Depenetration corridor creation is similar as for penetration corridors except that links must be established between corridor exit and permissable recovery bases. Attributes MYRECOV1, MYRECOV2, MYRECOV3, and MYRECOV4 define the DESIG of the recovery bases where an aircraft may land if exiting a given corridor. No other action is required for linkage. Commands are:

```
CREATE SETTING CLASS=DEPCOR SIDE=BLUE
       (CORNUM,MYRECOV1,MYRECOV2,MYRECOV3,MYRECOV4)=
       (1,AB800,AB801.AB802,AB803) &
       (2,AB804,AB805,AB806,AB807) &
       (3,AB500,AB501,AB502,AB503) &
       (4,AB504,'0','0','0') &
       (5,AB505,AB506,'0','0')

CREATE SETTING CLASS=DEPCOR SIDE=BLUE
       (CORNUM,DOGLEG,LAT,LONG)=
       (1,1,43,222) & (1,2,43,222) &
       (2,1,38,227) & (2,2,38,227) &
       (3,1,42,331) & (3,2,42,331) &
       (4,1,42,320) & (4,2,42,320) &
       (5,1,37,305) & (5,2,37,305) &
       (6,1,37,288) & (6,2,37,288)
```

Refuel points are created through a simple definition of latitudes and longitudes.

Recovery Bases. Recovery bases may be identified by the user at any point after the targets which represent them are stored in the data base. The process is to create a record by specifying a value for CAPACITY. In the create statement the user must be aware of the necessity of adequately defining the type of the target along with its DESIG. This usually means that the type's name (TYPE) and location (CNTRYLOC) be supplied with the DESIG. An example is:

```
CREATE SETTING SIDE=RED CLASS=AIRFIELD TYPE=RECOV CAPACITY=40
       (DESIG,CNTRYLOC)=(AB516,IT) & (AB513,EG) & (AB514,EG) &
       (AB515,EG) & (AB512,PO) & (AB511,CZ) & (AB509,YG) &
       (AB510,YG) & (AB505,BG) & (AB506,BG) & (AB507,BG) &
       (AB508,BG) & (AB155,UR) & (AB149,UR) & (AB150,UR) &
       (AB151,UR) & (AB152,UR) & (AB146,CU) & (AB147,CU) &
       (AB148,CU) & (AB153,MX) & (AB154,MX) & (AB145,MX)
```

APPENDIX F

EXECUTION OF THE QUICK SYSTEM

This appendix presents the detailed characteristics of user procedures for executing the QUICK system or more appropriately the execution of QUICKs executive software, the COP. The capability enables the user to submit batch-mode jobs or to execute remote terminal jobs.

These capabilities and the necessary data file definition are further described in the following subsections.

BATCH

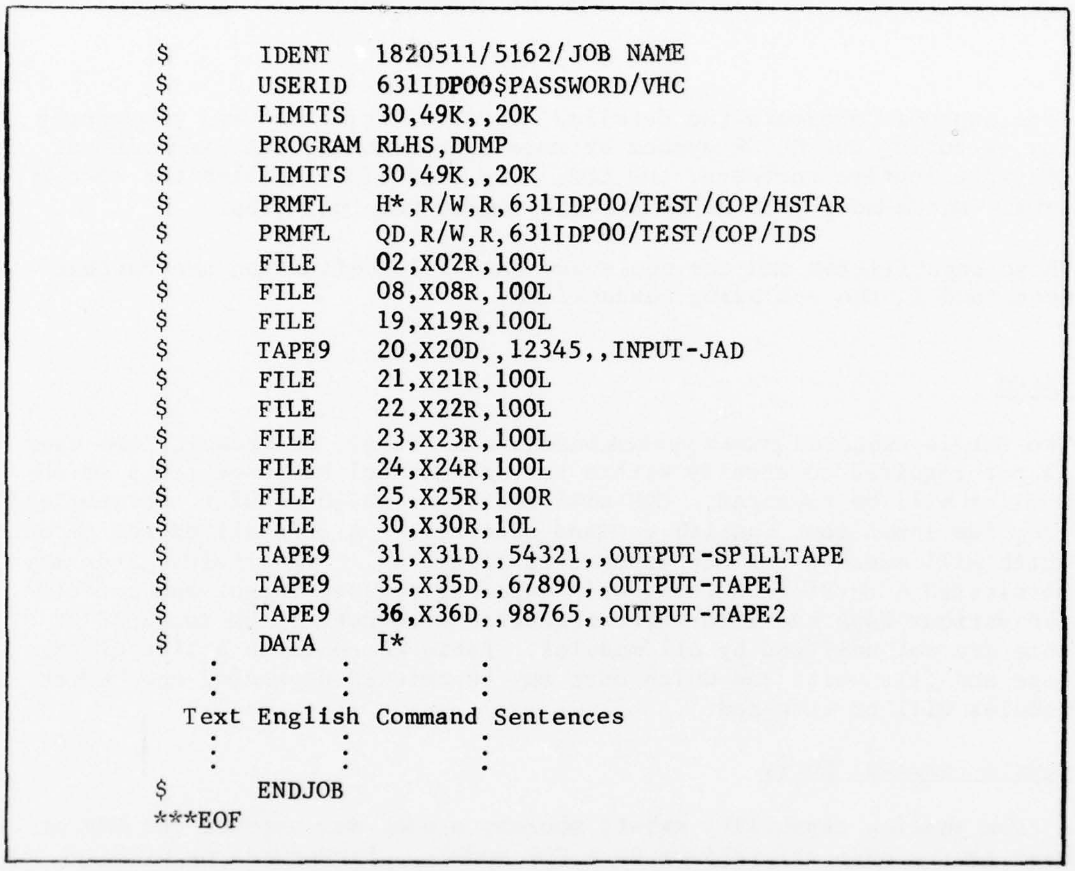
The COP is executed from a system loadable H* file. As a result, the user is not required to specify within the Job Control Language (JCL) which modules will be executed. COP will determine which modules are required from the input text English command sentences. A generalized set of JCL which will execute the COP appears in figure 25. The limits given may be altered according to the user's expected printed output and run time. The various tape *and file units requested* need not all be included as some are not utilized by all modules. Table 22 contains a list of the tape and file units and which ones may be omitted depending upon which modules will be executed.

Remote Terminal Entry

A time sharing capability exists whereby a user may execute the COP or make source code corrections to a COP module. Procedures to utilize this function are outlined below. In the following narrative, commands to be entered by the user will be flagged by an arrowhead (►).

Step 1. Log On and Initiation. The first step involved is to log on to the HIS system and to specify what operating system is to be used and, then, what modules are to be executed. Steps involved in this example utilizes a catalog file called 631IDPOO/COPRUN which contains the procedures to execute the COP. These steps are:

```
► $*$LOG24,TSS
  TERMINAL OK
  USERID$PASSWORD
►       (enter userid and password)
  IDENT?
►       (enter ident card information)
  X BLOCKS FILE SPACE AVAILABLE
  CLASSIFICATION OF YOUR INPUT?
►       (enter classification code, i.e., UNCL)
  CLASSIFICATION OF ITEMS YOU WILL CREATE?
►       (enter classification code, i.e., UNCL)
```



```
$       IDENT   1820511/5162/JOB NAME
$       USERID  631IDP00$PASSWORD/VHC
$       LIMITS  30,49K,,20K
$       PROGRAM RLHS,DUMP
$       LIMITS  30,49K,,20K
$       PRMFL   H*,R/W,R,631IDP00/TEST/COP/HSTAR
$       PRMFL   QD,R/W,R,631IDP00/TEST/COP/IDS
$       FILE    02,X02R,100L
$       FILE    08,X08R,100L
$       FILE    19,X19R,100L
$       TAPE9   20,X20D,,12345,,INPUT-JAD
$       FILE    21,X21R,100L
$       FILE    22,X22R,100L
$       FILE    23,X23R,100L
$       FILE    24,X24R,100L
$       FILE    25,X25R,100R
$       FILE    30,X30R,10L
$       TAPE9   31,X31D,,54321,,OUTPUT-SPILLTAPE
$       TAPE9   35,X35D,,67890,,OUTPUT-TAPE1
$       TAPE9   36,X36D,,98765,,OUTPUT-TAPE2
$       DATA    I*
    :       :         :
    :       :         :
   Text English Command Sentences
    :       :         :
    :       :         :
$       ENDJOB
***EOF
```

Figure 28. Batch-mode JCL

Table 22. Batch-mode JCL File Utilization

Unit File Code	Comment
H*	Must be included; contains system loadable COP modules.
QD	Must be included; contains the COP IDS data base file
02	TGTFIL used by IIM
08	BASFIL used by IIM
15	Weapon data file for ALOC
18	BTB Tape, JAD output (ASTERISK option)
19	TARFIL used by IIM
20	JAD input unit. Used by JLM
21	Internal sort files. Used by: JLM, REPORT, EIM, EVALUATE, PLANSET
22	
23	
24	
25	Random storage file used in concert with sort: JLM, REPORT, EIM, INDEXER
30	Directory file, used by IIM
31	Spill tape, used by IIM
32	Tape with IDS data base for SAVE or RESTORE (SRM)
35	Output tape files, used by EIM
36	

```
     SYSTEM ?
  ► YFORT 0 631IDP00/COPRUN
     READY
     *
  ► RUN COPRUN
```

The user now is interacting with a YFORT TSS subsystem program. The YFORT program will signal its desire for a response by outputting an equal sign (=) to which the user responds accordingly.

After the system outputs an equal sign, the user has multiple responses in some cases. In order to assist the user, a response of HELP may be entered and the system will respond with all the possible replies. Permissible use of the HELP command is outlined below.

The first question asked is whether or not output is to be displayed at a terminal. The following message is displayed:

```
     DO YOU WANT THE OUTPUT DIRECTED TO JOUT?  (YES OR NO)
```

A YES response will permit the use of SYSTEM JOUT. Otherwise all output will be directed to the printer.

Step 2. Mode Selection. The YFORT program is asking if the user desires to execute the COP, update the source, or both by displaying the command:

```
     ENTER MODE (RUN, COMPILE)
     =
```

If a run only is desired enter RUN and skip to step 4. If modules are to be recompiled first, enter

```
  ► COMPILE
```

Two other modes also exist, INITIALIZE and OBJECT. INITIALIZE will cause the IDS data file to be reinitialized before the current run. The MODE question is repeated. OBJECT will permit the user to execute using only object decks and a temporary H*.

Step 3. Compiling an Updated COP. The user is compiling module(s) and the program asks which one(s) by displaying:

```
     WHICH MODULES?
```

The names of the modules should be separated by commas with no imbedded blanks. HELP will produce a list of the legal modules and a repeat of the question.

A possible response would be:

```
  [illegible]
```

In order to ascertain if this is a compilation and execution, the program displays:

```
ENTER RUN MODE?  (YES, NO)
=
```

If the user wishes to compile only, enter NO and skip to step 7; otherwise enter YES.

The OBJECT mode differs slightly from the COMPILE mode. The first question asked is:

```
WHICH MODULES DO YOU WANT TO RECOMPILE?
```

The names of the modules should be entered, separated by commas and with no imbedded blanks.

The next message is:

```
WHAT OTHER MODULES ARE NEEDED?
```

Any modules that are needed for the run but do not need to be recompiled are entered now. The final difference between the two modes is that the option to bypass the *RUN* mode *is not given.*

<u>Step 4. Limit Specification</u>. The COP executes a wide variety of modules and differing size data bases which implies varying limits as to computer resource usage in terms of computer time, core storage, and lines to be printed. A series of commands are given to provide interactive capability whereby the limits may be altered. First, the system must know which modules (or verbs) to execute. Hence the command:

```
WHICH VERBS
```

The verbs are separated by commas with no imbedded blanks. If HELP is entered the list of valid verbs is displayed and the question is asked again.

A possible response is:

```
► CREATE,CHANGE,PRINT
```

The program responds with a set of default limits such as

```
LIMITS ARE [illegible]
NEW LIMITS? (ENTER NO OR ENTER NEW LIMITS)
=
```

If the displayed limits are valid, a NO response is entered. Otherwise new limits are entered. For example:

```
► [illegible]
```

Step 5. Text English Command Entry. User inputs are now given. The display command is:

```
        INPUT DATA FILE (ENTER CATALOG/FILE STRING OR DONE)
        =
```

Here the program is asking the user to name a data file in which the user has stored text English commands. The user will be able to enter any number of such files, in the order as input. The user will also be able to enter text English commands directly and interweave files with direct entry by following this procedure. If the user enters the catalog file string of a file, the program will return to the beginning of step 5. If the user has entered all the files names desired, DONE or a blank is entered.

The next message displayed is:

```
        ANY ADDITIONAL DATA? (YES,NO)
```

A NO response will terminate step 5.

The program now requests the possibility of text English commands entered directly by displaying:

```
        ENTER DATA CARDS (BLANK LINE TERMINATES)
        =
```

All inputs are entered followed by a blank line.

To allow for the possibility of additional data, the program displays;

```
        ANY ADDITIONAL DATA FILES (YES,NO)
```

If YES is entered, the program returns to the beginning of step 5. A NO response terminates step 5.

Step 6. Special Files. Certain verbs may require special input or output files. For each verb entered in step 4 that may require a special file the following type of message is displayed:

```
        THE SELECT VERB MAY NEED [illegible] FORMAT INPUT FILE
        ENTER THE FILE STRING ONE OF THE FOLLOWING FORMATS
        IS BLANK OR NULL RESPONSE IF THE FILE IS NOT NEEDED.
        [illegible]
        [illegible]
        [illegible]
```

The proper file [illegible] should be [illegible] based on the [illegible] of [illegible]

Step 7. Executing the COP. The constructed job may now be submitted which must be done within the CARDIN subsystem of HIS. The following is displayed:

```
                                    COMPILE COP
     THE JOB HAS BEEN BUILT TO RUN COP
                                    COMPILE AND RUN COP
     AND DIRECT THE OUTPUT TO JOUT
     TO EXECUTE ENTER-
     RUN THE JOB

     READY
     *
  ➤  RUN THE JOB

     SNUMB # XXXXX
     *
```

The job has now been entered, the user should note the SNUMB.

Preparing an IDS File for COP

The COP must run on an IDS file. When the user wishes to restore onto a previously unused file or build a QUICK data base from scratch, he must use a file specially prepared for IDS. The creation of such a file has two steps. First, the user must create the file using the File Management Subsystem (FMS). Besides the usual options employed to create a random file, the following additional options are used:

BASESIZE/N/ - N defines the maximum number of pages in the IDS data file

RNG/r1,r2/ - Defines the page range (for QUICK applications set r1=1 and r2=N as above)

LINESPERPAGE/m/ - m defines the number of lines on an IDS page (for QUICK applications set m=21)

An example of an appropriate FCREATE directive would be:

```
FCREATE/IDS/ MYFILE, BASESIZE/401/,RNG/1,401/,LINESPERPAGE/21/,
                SIZE/102/,MODE/RAND/,FCLASS/UZZ/
```

Following the creation of the IDS file, the user must initialize the file by utilizing an IDS utility called QUTI. This utility has a single input card that specifies the page range to be initialized. An example of an appropriate QUTI activity would be:

```
$     PROGRAM  QUTI
$     PRMFL    A1,R/W,R,631IDP00/MYFILE
IDS   INITIAL  1,401
```

If the user needs any further assistance, consult Honeywell Reference DC53A, Rev.0 #I-D-S/I USER'S GUIDE.

IDS File Unlock

If a run of the QUICK system aborts during processing, very often the user will be informed that the IDS file is abort locked. This message will appear in the print report. This condition must be corrected before any further processing can occur. Therefore, the user must 'unlock' the IDS data file. This is done via a FILSYS activity which contains a directive similar to the following:

```
ALOCK    631IDP00/TEST/COP/IDS,OFF
```